

Current KAS KNART

Package in package 'ConceptualKnart'

Current KAS KNART
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/13/2017. Last modified 2/23/2018

Action Groups diagram

Class diagram in package 'Current KAS KNART'

Action Groups
Version 1.0
lconstab created on 10/18/2017. Last modified 2/23/2018

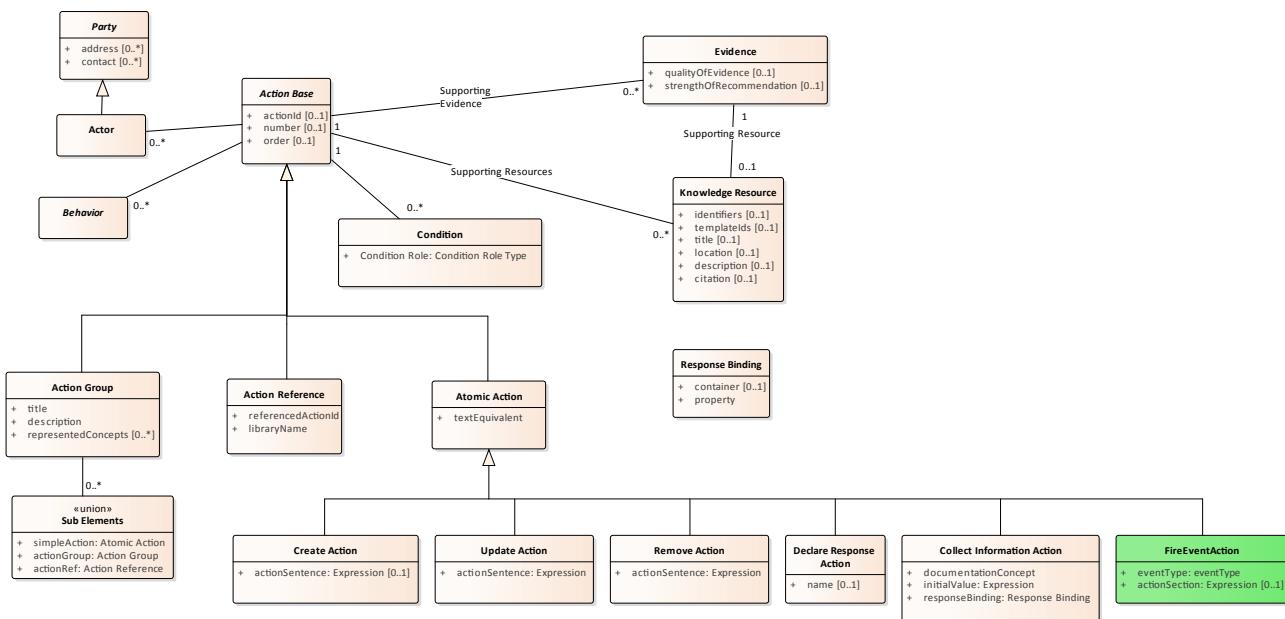


Figure 1: Action Groups

Action Base

Class in package 'Action Group'

Actions are the output of the CDS system and represent the tasks that must be carried out by a human or a computer system.

Action Base
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS
<p>⇒ Generalization from Action Group to Action Base</p> <p>[Direction is 'Source -> Destination'.]</p>
<p>⇒ Generalization from Atomic Action to Action Base</p> <p>[Direction is 'Source -> Destination'.]</p>

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from Action Reference to Action Base	[Direction is 'Source -> Destination'.]
ATTRIBUTES	
◆ actionId : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	An identifier for the action. The identifier must be unique within the scope of the artifact. [Is static False. Containment is Not Specified.]
◆ number : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	A number for the action primarily for display to the user. This is different than the actionId which is used as a key to identify the action or group internally. [Is static False. Containment is Not Specified.]
◆ order : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	If specified, determines the ordering for this element within the group. [Is static False. Containment is Not Specified.]
ASSOCIATIONS	
↙ Association (direction: Unspecified) Supporting Evidence	The evidence grade and the sources of evidence associated with this artifact.
Source: Public (Class) Action Base Cardinality: [1]	Target: Public (Class) Evidence Cardinality: [0..*]
↙ Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Actor Cardinality: [0..*]
↙ Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Behavior Cardinality: [0..*]
↙ Association (direction: Unspecified) Supporting Resources	Didactic or other informational resources associated with the action that can be provided to the CDS recipient. Information resources can include inline text commentary and links to web resources.
Source: Public (Class) Action Base Cardinality: [1]	Target: Public (Class) Knowledge Resource Cardinality: [0..*]

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Condition Cardinality: [0..*]

Action Group

Class in package 'Action Group'

This type of action is used to organize a group of related actions into one container. The semantics of how the group's subelements interact with each other and how the subelements might be presented are specified in the group behavior.

Action Group
 Version 1.0 Phase 1.0 Proposed
 Iconstab created on 10/18/2017. Last modified 11/1/2017
 Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS	
 Generalization from Action Group to Action Base	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
 title : Public	A brief title that is shown to the user of the artifact, i.e., the recipient of the CDS [Is static False. Containment is Not Specified.]
 description : Public	A lengthier description of the action group that can be displayed to the user or the recipient of the CDS [Is static False. Containment is Not Specified.]
 representedConcepts : Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	The concept(s) represented by this action group. For instance, a section may represent a group of beta-blockers, a composite orderable such as an insulin sliding scale, or a set of order sentences for a particular orderable (e.g., Tylenol). [Is static False. Containment is Not Specified.]

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) Sub Elements «union» Cardinality: [0..*]	Target: Public (Class) Action Group
 Association (direction: Unspecified)	

ASSOCIATIONS	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Group Cardinality: [1]

Action Reference

Class in package 'Action Group'

This subelement specifies an action to be included by reference from a referenced library as defined in the libraries section of the metadata

A reference to an action defined in a library.

Action Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/30/2017. Last modified 11/1/2017
Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Action Reference to Action Base	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ referencedActionId : Public	
The Id of the action being referenced.	[Is static False. Containment is Not Specified.]
◆ libraryName : Public	[Is static False. Containment is Not Specified.]

Actor

Class in package 'Action Group'

The actor that is responsible for executing the action. This element is used when an artifact can have multiple actors responsible for the execution of various CDS actions, for example, an interdisciplinary plan of care.

It is important to note the distinction between a CDS action and an ensuing clinical action. A CDS action might be to order patient ambulation, the actor for which is a physician responsible for writing the order. A nurse might be responsible for ensuring that the patient ambulates. In this case, the artifact will specify the physician as the actor.

Actor
Version 1.0 Phase 1.0 Proposed
Iconstab created on 11/7/2017. Last modified 11/7/2017
Extends Party

OUTGOING STRUCTURAL RELATIONSHIPS	

OUTGOING STRUCTURAL RELATIONSHIPS	
⬅ Generalization from Actor to Party	[Direction is 'Source -> Destination'.]

ASSOCIATIONS	
✍ Association (direction: Unspecified)	

Source: Public (Class) Action Base Target: Public (Class) Actor
Cardinality: [0..*]

Atomic Action

Class in package 'Action Group'

The subelement is an atomic or single action.

An action that is not further broken down into constituent actions.

Atomic Action
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 11/1/2017
Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS	
⬅ Generalization from Atomic Action to Action Base	[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS	
➡ Generalization from Create Action to Atomic Action	[Direction is 'Source -> Destination'.]
➡ Generalization from Remove Action to Atomic Action	[Direction is 'Source -> Destination'.]
➡ Generalization from FireEventAction to Atomic Action	[Direction is 'Source -> Destination'.]
➡ Generalization from Update Action to Atomic Action	[Direction is 'Source -> Destination'.]
➡ Generalization from Declare Response Action to Atomic Action	[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from Collect Information Action to Atomic Action	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ textEquivalent : Public A brief textual description of the action that summarizes the action	[Is static False. Containment is Not Specified.]

Collect Information Action

Class in package 'Action Group'

This action requests information from the actor. The information request is specified as a DocumentationItem.

Collect Information Action
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS	
◀ Generalization from Collect Information Action to Atomic Action	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ documentationConcept : Public This provides a specification of the information to be collected from the user.	[Is static False. Containment is Not Specified.]
◆ initialValue : Expression Public An expression to compute an initial value for the documentation concept. The initial value could be computed from previous data about the patient available via expressions specified in the external data	[Is static False. Containment is Not Specified.]
◆ responseBinding : Response Binding Public Defines the response binding for the documentation item.	[Is static False. Containment is Not Specified.]

Create Action

Class in package 'Action Group'

A new action to be executed by a user or a computer system. The sentence provides the details of the action to be executed.

Create Action
 Version 1.0 Phase 1.0 Proposed
 Iconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Create Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ actionSentence : Expression Public
 Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The parameters of the action that is to be executed. For example, an action may be to order a medication.

[Is static False. Containment is Not Specified.]

Declare Response Action

Class in package 'Action Group'

The DeclareResponseAction provides a mechanism to declare a container for responses provided by the user in response to CollectInformationActions. The intended semantics are to provide a container that can be used to access responses within expressions used subsequently in an artifact.

The DeclareResponseAction creates a named container within the Parameters scope of the artifact, and expressions may access the contents of a response using a ParameterRef expression. The container is expected to be a collection of name-value pairs, and the intended semantics are to allow the Property expression to be used, in connection with a ParameterRef expression as the source, to retrieve the current value for a property.

The CollectInformationAction contains a responseBinding attribute that specifies the name of the container, and the name of the property to be used to store the response value.

If no Name attribute is provided, the response container will be named Responses.

Declare Response Action
 Version 1.0 Phase 1.0 Proposed
 Iconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Declare Response Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public
 Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Evidence

Class in package 'Action Group'

Reference to research on which the artifact is based. This evidence can be 'graded' depending on its quality and pedigree and the strength of the recommendations it makes.

Evidence
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 10/30/2017

ATTRIBUTES

- ◆ qualityOfEvidence : Public
Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The quality of the evidence associated with this artifact. The codeSystem attribute specifies the quality scale used to grade this evidence source while the code specifies the actual quality score (represented as a coded value) associated with this evidence reference. CodeSystemName specifies the name of the scale. DisplayName specifies the display name of the coded value (the score).

[Is static False. Containment is Not Specified.]

- ◆ strengthOfRecommendation : Public
Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The strength of the recommendation assigned to this reference. The codeSystem attribute specifies the scale used to grade this evidence source while the code specifies the actual score (represented as a coded value) for the strength of the evidence. CodeSystemName specifies the name of the scale. DisplayName specifies the display name of the coded value (the score).

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- ✓ Association (direction: Unspecified) Supporting Resource

Source: Public (Class) Evidence
Cardinality: [1]

Target: Public (Class) Knowledge Resource
Cardinality: [0..1]

- ✓ Association (direction: Unspecified) Supporting Evidence

The evidence grade and the sources of evidence associated with this artifact.

Source: Public (Class) Action Base
Cardinality: [1]

Target: Public (Class) Evidence
Cardinality: [0..*]

FireEventAction

Class in package 'Action Group'

This action fires an event. The event can serve as a trigger to another artifact.

FireEventAction
Version 1.0 Phase 1.0 Proposed
Iconstab created on 2/23/2018. Last modified 2/23/2018
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS	
<p>↳ Generalization from FireEventAction to Atomic Action</p>	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
<p>◆ eventType : eventType Public</p> <p>The type of the event that is fired.</p>	[Is static False. Containment is Not Specified.]
<p>◆ actionSection : Expression Public</p> <p>Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>The action sentence is the payload of the event. Another artifact receives this payload as an input.</p>	[Is static False. Containment is Not Specified.]

Knowledge Resource

Class in package 'Action Group'

KnowledgeResource specifies a reference to an associated resource of relevance to the artifact such as a guideline, a performance measure, another knowledge artifact, or a source of evidence for the artifact.

Knowledge Resource
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 10/30/2017

ATTRIBUTES	
<p>◆ identifiers : Public</p> <p>Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>The set of unique identifiers for this resource.</p>	[Is static False. Containment is Not Specified.]
<p>◆ templateIds : Public</p> <p>Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>The set of unique identifiers for the templates associated with this resource.</p>	[Is static False. Containment is Not Specified.]
<p>◆ title : Public</p> <p>Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>The title of the document</p>	[Is static False. Containment is Not Specified.]
<p>◆ location : Public</p>	

ATTRIBUTES	
Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	
The URL of the given resource.	[Is static False. Containment is Not Specified.]
◆ description : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	
A short textual description of the resource.	[Is static False. Containment is Not Specified.]
◆ citation : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	
The resource citation.	[Is static False. Containment is Not Specified.]

ASSOCIATIONS	
✓ Association (direction: Unspecified) Supporting Resource	
Source: Public (Class) Evidence Cardinality: [1]	Target: Public (Class) Knowledge Resource Cardinality: [0..1]
✓ Association (direction: Unspecified) Supporting Resources	
Didactic or other informational resources associated with the action that can be provided to the CDS recipient. Information resources can include inline text commentary and links to web resources.	
Source: Public (Class) Action Base Cardinality: [1]	Target: Public (Class) Knowledge Resource Cardinality: [0..*]

Party

Class in package 'Action Group'

Party
Version 1.0 Phase 1.0 Proposed
lconstab created on 11/7/2017. Last modified 11/7/2017

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from Actor to Party	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ address : Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	
An individual address of type AD associated with this entity.	

ATTRIBUTES	
	[Is static False. Containment is Not Specified.]
<p>◆ contact : Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)</p> <p>An individual contact item of type TEL associated with this entity.</p>	
	[Is static False. Containment is Not Specified.]

Remove Action

Class in package 'Action Group'

This action removes another proposed action or an ongoing action.

Remove Action
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Remove Action to Atomic Action	[Direction is 'Source -> Destination'.]
ATTRIBUTES	
◆ actionSentence : Expression Public The expression must resolve to the action that is being removed.	[Is static False. Containment is Not Specified.]

Response Binding

Class in package 'Action Group'

Defines the attributes required to specify a binding path for documentation item responses. The container attribute specifies the name of the response container that will be used. If no container attribute is provided, the default container name of Responses will be used. The property attribute specifies the name of the property within the container that will be used to store the user response value.

Response Binding
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 11/1/2017. Last modified 11/1/2017

ATTRIBUTES	
◆ container : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

ATTRIBUTES

◆ property : Public

[Is static False. Containment is Not Specified.]

Update Action

Class in package 'Action Group'

This action changing the value of another existing action. The action being modified may be a proposed action (e.g., an medication being prescribed by a clinician) or be an ongoing action (e.g., an existing prescription). In these cases, a modification can be used to change the dose of the medication. It may also be used to discontinue a medication by changing the stop date.

Update Action
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Update Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ actionSentence : Expression Public

The modification to the action. This is specified by modifying the properties of an existing action using an ObjectRedefine expression.

[Is static False. Containment is Not Specified.]

Behavior

Class in package 'Behaviors'

The behaviors associated with how the action is presented and executed. The semantics and the validity of behaviors for actions are described elsewhere.

A behavior may be specified for a specific action or a group of actions. This is the base type for all Behaviors.

Behavior
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Group Selection Behavior to Behavior

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from Precheck Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Cardinality Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Required Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Visual Style Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Group Organization Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Read Only Behavior to Behavior	[Direction is 'Source -> Destination'.]

ASSOCIATIONS	
✍ Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Behavior Cardinality: [0..*]

Condition

Class in package 'Conditions'

The conditions section lists all conditions that pertain to the action. Conditions define the logic that determine the applicability of the action in the given context, any precondition or post condition, and/or any inclusion and exclusion criteria for the given action.

A condition specifies when a knowledge component is to be executed. For example, an ECA rule uses an ApplicableScenario condition to determine whether or not the action described by the artifact should be executed.

A collection of conditions that are used to define whether various aspects of the artifact, such as whether or not a particular action should be executed, or whether a particular order set item is applicable to a given patient

Condition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES	
◆ Condition Role : Condition Role Type Public	

ATTRIBUTES

The role determines when to evaluate the expression and how to proceed based on the expression results. Different artifact types use different types of conditions to control various aspects of the artifact. See the condition role type enumeration documentation for more discussion.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- ✓ Association (direction: Unspecified)

Source: Public (Class) Action Base

Target: Public (Class) Condition

Cardinality: [0..*]

Sub Elements

Class «union» in package 'Metadata'

The constituent elements of the group can be of different types including subgroups, simple or atomic actions, and subgroups embedded by reference. While the group allows artifacts of different types to be mixed and matched in any combination, a particular type of artifact might further restrict the combinations.

For example, an artifact type might require subelements of a particular group to be either groups or simple actions; elements of both types cannot exist in the group.

Sub Elements

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/30/2017. Last modified 11/1/2017

ATTRIBUTES

- ◆ simpleAction : Atomic Action Public

[Is static False. Containment is Not Specified.]

- ◆ actionGroup : Action Group Public

[Is static False. Containment is Not Specified.]

- ◆ actionRef : Action Reference Public

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- ✓ Association (direction: Unspecified)

Source: Public (Class) Sub Elements «union»

Target: Public (Class) Action Group

Cardinality: [0..*]

Behaviors diagram

Class diagram in package 'Current KAS KNART'

Behaviors
Version 1.0

Iconstab created on 10/31/2017. Last modified 10/31/2017

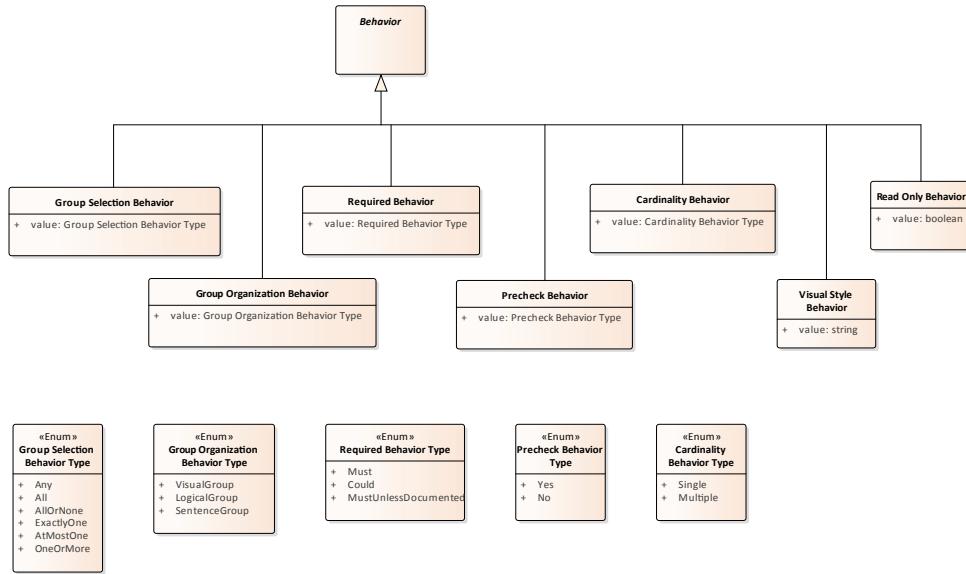


Figure 2: Behaviors

Behavior

Class in package 'Behaviors'

The behaviors associated with how the action is presented and executed. The semantics and the validity of behaviors for actions are described elsewhere.

A behavior may be specified for a specific action or a group of actions. This is the base type for all Behaviors.

Behavior

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS
<p>⇒ Generalization from Group Selection Behavior to Behavior</p> <p>[Direction is 'Source -> Destination'.]</p>
<p>⇒ Generalization from Precheck Behavior to Behavior</p> <p>[Direction is 'Source -> Destination'.]</p>
<p>⇒ Generalization from Cardinality Behavior to Behavior</p> <p>[Direction is 'Source -> Destination'.]</p>
<p>⇒ Generalization from Required Behavior to Behavior</p> <p>[Direction is 'Source -> Destination'.]</p>

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from Visual Style Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Group Organization Behavior to Behavior	[Direction is 'Source -> Destination'.]
⇒ Generalization from Read Only Behavior to Behavior	[Direction is 'Source -> Destination'.]

ASSOCIATIONS	
 Association (direction: Unspecified) Source: Public (Class) Action Base	Target: Public (Class) Behavior Cardinality: [0..*]

Cardinality Behavior

Class in package 'Behaviors'

For a group or an action, specifies if that item may be repeated.

Cardinality Behavior
 Version 1.0 Phase 1.0 Proposed
 Iconstab created on 10/31/2017. Last modified 11/1/2017
 Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS	
← Generalization from Cardinality Behavior to Behavior	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
♦ value : Cardinality Behavior Type Public	[Is static False. Containment is Not Specified.]

Cardinality Behavior Type

Class «Enum» in package 'Behaviors'

Defines behavior for an action or a group for how many times that item may be repeated, i.e., cardinality. For example, if a user is documenting lesions, the lesion element may be repeated several times, once for each occurrence of a lesion on the patient or tissue sample or image

Cardinality Behavior Type
 Version 1.0 Phase 1.0 Proposed
 Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ Single : Public</p> <p>An action with this behavior is one of the most frequent actions that is, or should be, included by an end user, for the particular context in which the action occurs. The system displaying the action to the end user should consider "pre-checking" such an action as a convenience for the user.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Multiple : Public</p> <p>An action with this behavior is one of the less frequent actions included by the end user, for the particular context in which the action occurs. The system displaying the actions to the end user would typically not "pre-check" such an action.</p> <p>[Is static False. Containment is Not Specified.]</p>

Group Organization Behavior

Class in package 'Behaviors'

For a group of actions, specifies the organizational intent of the grouping. This is meant to provide a hint to the system which displays the group of actions to an end user.

Group Organization Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS
<p>↳ Generalization from Group Organization Behavior to Behavior</p> <p>[Direction is 'Source -> Destination'.]</p>
ATTRIBUTES
<p>◆ value : Group Organization Behavior Type Public</p> <p>[Is static False. Containment is Not Specified.]</p>

Group Organization Behavior Type

Class «Enum» in package 'Behaviors'

Defines organization behavior of a group: gives the reason why the items are grouped together.

Group Organization Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ VisualGroup : Public</p> <p>Any group marked with this behavior should be displayed as a visual group to the end user.</p>

ATTRIBUTES	
	[Is static False. Containment is Not Specified.]
 LogicalGroup : Public	
A group with this behavior logically groups its sub-elements, and may be shown as a visual group to the end user, but it is not required to do so.	[Is static False. Containment is Not Specified.]
 SentenceGroup : Public	
A group of related alternative actions is a sentence group if the item referenced by the action is the same in all the actions, and each action simply constitutes a different variation on how to specify the details for that item. For example, two actions that could be in a SentenceGroup are "aspirin, 500 mg, 2 times per day" and "aspirin, 300 mg, 3 times per day". In both cases, aspirin is the item referenced by the action, and the two actions represent two different options for how aspirin might be ordered for the patient. Note that a SentenceGroup would almost always have an associated selection behavior of "AtMostOne", unless it's a required action, in which case, it would be "ExactlyOne".	[Is static False. Containment is Not Specified.]

Group Selection Behavior

Class in package 'Behaviors'

For a group of actions, specifies the number of actions that may be chosen by an end user.

Group Selection Behavior
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS	
 Generalization from Group Selection Behavior to Behavior	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
 value : Group Selection Behavior Type Public	[Is static False. Containment is Not Specified.]

Group Selection Behavior Type

Class «Enum» in package 'Behaviors'

Defines selection behavior of a group: specifies the number of selectable items in the group that may be selected by the end user when the items of the group are displayed.

Group Selection Behavior Type
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES	
------------	--

ATTRIBUTES	
◆ Any : Public	[Is static False. Containment is Not Specified.]
◆ All : Public	[Is static False. Containment is Not Specified.]
◆ AllOrNone : Public	[Is static False. Containment is Not Specified.]
◆ ExactlyOne : Public	[Is static False. Containment is Not Specified.]
◆ AtMostOne : Public	[Is static False. Containment is Not Specified.]
◆ OneOrMore : Public	[Is static False. Containment is Not Specified.]

Precheck Behavior

Class in package 'Behaviors'

For a particular action, specifies how often the action is expected to be selected in the particular context of the group containing that action. In general, depending on the group selection behavior, there may be zero, one or more actions which are frequently selected. This setting can serve as a hint to the system that displays the action to the end user: some systems will pre-select those actions which are (or should be) most frequently selected.

Precheck Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Precheck Behavior to Behavior	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ value : Precheck Behavior Type Public	[Is static False. Containment is Not Specified.]

Precheck Behavior Type

Class «Enum» in package 'Behaviors'

Defines selection frequency behavior for an action or group; i.e., for most frequently selected items, the end-user system may provide convenience options in the UI (such as pre-selection) in order to (1) communicate to the end user what the most frequently selected item is, or should, be in a particular context, and (2) save the end user time.

Precheck Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

- ◆ Yes : Public

An action with this behavior is one of the mos frequent actions that is, or should be, included by an end user, for the particular context in which the action occurs. The system displaying the action to the end user should consider "pre-checking" such an action as a convenience for the user.

[Is static False. Containment is Not Specified.]

- ◆ No : Public

An action with this behavior is one of the less frequent actions included by the end user, for the particular context in which the action occurs. The system displaying the actions to the end user would typically not "pre-check" such an action.

[Is static False. Containment is Not Specified.]

Read Only Behavior

Class in package 'Behaviors'

For a particular action or action group, specifies whether the elements are read only.

Read Only Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

- ◀ Generalization from Read Only Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

- ◆ value : boolean Public

[Is static False. Containment is Not Specified.]

Required Behavior

Class in package 'Behaviors'

For a single action, specifies what level of requiredness is associated with the action.

Required Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Required Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Required Behavior Type Public

[Is static False. Containment is Not Specified.]

Required Behavior Type

Class «Enum» in package 'Behaviors'

Defines requiredness behavior for selecting an action or an action group; i.e., whether the action or action group is required or optional.

Required Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

◆ Must : Public

An action with this behavior must be included in the actions processed by the end user; the end user may not choose not to include this action.

[Is static False. Containment is Not Specified.]

◆ Could : Public

An action with this behavior may be included in the set of actions processed by the end user.

[Is static False. Containment is Not Specified.]

◆ MustUnlessDocumented : Public

An action with this behavior must be included in the set of actions processed by the end user, unless the end user provides documentation as to why the action was not included.

[Is static False. Containment is Not Specified.]

Visual Style Behavior

Class in package 'Behaviors'

For a group or an action, specifies the visual style for the action.

Visual Style Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Visual Style Behavior to Behavior	[Direction is 'Source -> Destination'.]
ATTRIBUTES	
◆ value : string Public	[Is static False. Containment is Not Specified.]

Conditions diagram

Class diagram in package 'Current KAS KNART'

Conditions
Version 1.0
Iconstab created on 10/31/2017. Last modified 2/23/2018

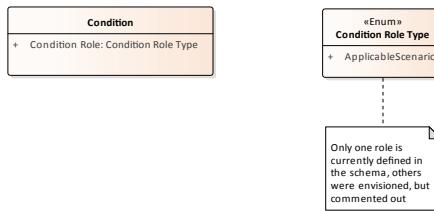


Figure 3: Conditions

Note

Note in package 'Current KAS KNART'

Only one role is currently defined in the schema, others were envisioned, but commented out

Note
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 10/31/2017
Extends

Condition

Class in package 'Conditions'

The conditions section lists all conditions that pertain to the action. Conditions define the logic that determine the applicability of the action in the given context, any precondition or post condition, and/or any inclusion and exclusion criteria for the given action.

A condition specifies when a knowledge component is to be executed. For example, an ECA rule uses an ApplicableScenario condition to determine whether or not the action described by the artifact should be executed.

A collection of conditions that are used to define whether various aspects of the artifact, such as whether or not a particular action should be executed, or whether a particular order set item is applicable to a given patient

Condition
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

- Condition Role : Condition Role Type Public

The role determines when to evaluate the expression and how to proceed based on the expression results. Different artifact types use different types of conditions to control various aspects of the artifact. See the condition role type enumeration documentation for more discussion.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Action Base

Target: Public (Class) Condition

Cardinality: [0..*]

Condition Role Type

Class «Enum» in package 'Conditions'

The roles that a condition plays in the execution of a component. Currently, only one role type is defined. Additional role types may be defined in the future (e.g., inclusion criteria, exclusion criteria)

Condition Role Type
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

- ApplicableScenario : Public

This role type specifies that a condition is used to determine whether or not a particular knowledge component should be executed. If the expression evaluates to true, then the component is executed.

[Is static False. Containment is Not Specified.]

Expressions diagram

Class diagram in package 'Current KAS KNART'

Expressions
 Version 1.0

lconstab created on 10/31/2017. Last modified 10/31/2017

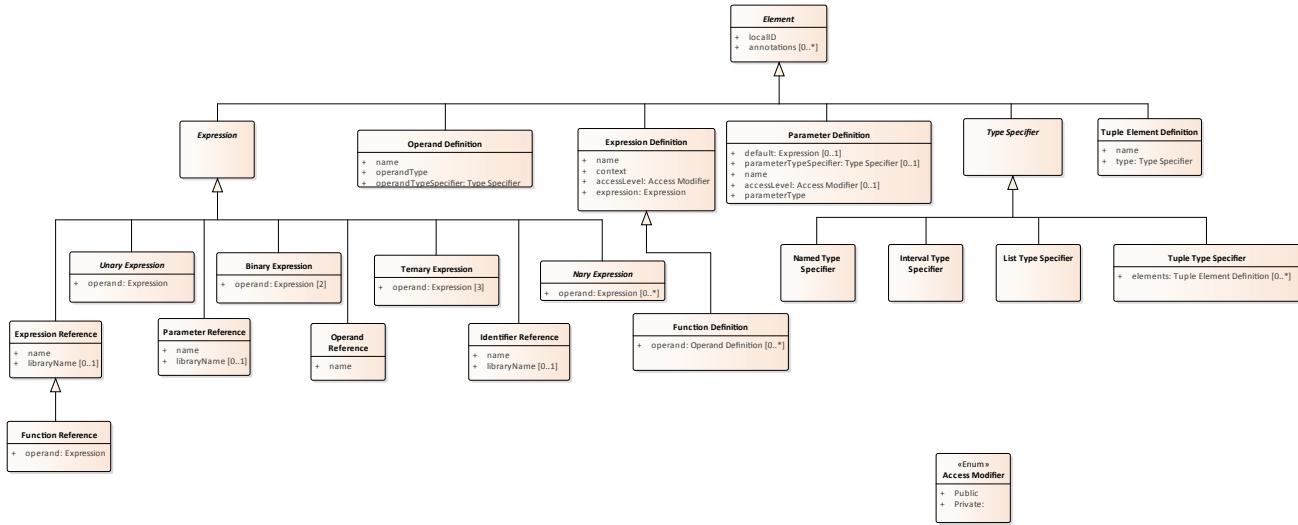


Figure 4: Expressions

Access Modifier

Class «Enum» in package 'Expressions'

The AccessModifier type is used to specify the access level for the various definitions within a library such as parameters, expressions, and functions. If no access modifier is specified, public is assumed. Private definitions can only be accessed within the library in which they are defined.

Access Modifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES	
◆ Public : Public	[Is static False. Containment is Not Specified.]
◆ Private : Public	[Is static False. Containment is Not Specified.]

Binary Expression

Class in package 'Expressions'

The BinaryExpression type defines the abstract base type for expressions that take two arguments.

Binary Expression
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Binary Expression to Expression	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ operand : Expression Public Multiplicity: ([2], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

Element

Class in package 'Expressions'

The Element type defines the abstract base type for all library elements in ELM.

Element
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from TypeSpecifier to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from ParameterDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from Expression to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from TupleElementDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from ExpressionDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from OperandDefinition to Element	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ localID : Public	[Is static False. Containment is Not Specified.]
◆ annotations : Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

The annotation element provides a mechanism for decorating expressions with application-specific information such as translation hints, visual designer information, or debug symbols.

[Is static False. Containment is Not Specified.]

Expression

Class in package 'Expressions'

The clinical (sub) scenario in which this action is applicable. This scenario is additive to any scenarios specified in the containers of this action such as action groups or the knowledge document. In other words, this scenario does not override the scenario constraints specified in the container elements.

The Expression type defines the abstract base type for all expressions used in the ELM expression language.

Expression
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Expression to Element

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

➡ Generalization from Parameter Reference to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Ternary Expression to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Expression Reference to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Operand Reference to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Unary Expression to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Identifier Reference to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Nary Expression to Expression

[Direction is 'Source -> Destination'.]

➡ Generalization from Binary Expression to Expression

[Direction is 'Source -> Destination'.]

Expression Definition

Class in package 'Expressions'

The named expression used to retrieve external data. For instance, an expression to retrieve patient demographic data or a set of SNOMED-CT codes subsumed by another SNOMED-CT code from a terminology server.

The ExpressionDef type defines an expression and an associated name that can be referenced by any expression in the artifact. The name must be unique within the artifact. The context attribute specifies the scope of the execution and is used by the environment to determine when and how to evaluate the expression.

Expression Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Expression Definition to Element

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

➡ Generalization from Function Definition to Expression Definition

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ context : Public

[Is static False. Containment is Not Specified.]

◆ accessLevel : Access Modifier Public

[Is static False. Containment is Not Specified.]

◆ expression : Expression Public

[Is static False. Containment is Not Specified.]

Expression Reference

Class in package 'Expressions'

The ExpressionRef type defines an expression that references a previously defined NamedExpression. The result of evaluating an ExpressionReference is the result of evaluating the referenced NamedExpression.

Expression Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Expression Reference to Expression

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Function Reference to Expression Reference

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ libraryName : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Function Definition

Class in package 'Expressions'

The FunctionDef type defines a named function that can be invoked by any expression in the artifact. Function names must be unique within the artifact. Functions may take any number of operands.

Function Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression Definition

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Function Definition to Expression Definition

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Operand Definition Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Function Reference

Class in package 'Expressions'

The FunctionRef type defines an expression that invokes a previously defined function. The result of evaluating each operand is passed to the function.

Function Reference

Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017
 Extends Expression Reference

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Function Reference to Expression Reference

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public

[Is static False. Containment is Not Specified.]

Identifier Reference

Class in package 'Expressions'

The IdentifierRef type defines an expression that references an unresolved identifier. An unresolved identifier reference generally indicates an error condition. The implementation is free to attempt to resolve the identifier, but is also justified in throwing an error at compile-time (or run-time for an interpretive system) when an identifier ref is encountered.

Identifier Reference

Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017
 Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Identifier Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ libraryName : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Interval Type Specifier

Class in package 'Expressions'

IntervalTypeSpecifier defines an interval type by specifying the point type. Any type can serve as the point type for an interval, so long as it supports comparison operators, minimum and maximum value determination, as well as predecessor and successor functions.

Interval Type Specifier
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Interval Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

List Type Specifier

Class in package 'Expressions'

ListTypeSpecifier defines a list type by specifying the type of elements the list may contain.

List Type Specifier
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from List Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

Named Type Specifier

Class in package 'Expressions'

NamedTypeSpecifier defines a type identified by a name, such as Integer, String, Patient, or Encounter.

Named Type Specifier
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Named Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

Nary Expression

Class in package 'Expressions'

The NaryExpression type defines an abstract base class for an expression that takes any number of arguments, including zero.

Nary Expression
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Nary Expression to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Operand Definition

Class in package 'Expressions'

The OperandDef type defines an operand to a function that can be referenced by name anywhere within the body of a function definition.

Operand Definition

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Operand Definition to Element

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ operandType : Public

[Is static False. Containment is Not Specified.]

◆ operandTypeSpecifier : Type Specifier Public

[Is static False. Containment is Not Specified.]

Operand Reference

Class in package 'Expressions'

The OperandRef expression allows the value of an operand to be referenced as part of an expression within the body of a function definition.

Operand Reference

Version 1.0 Phase 1.0 Proposed

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Operand Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

Parameter Definition

Class in package 'Expressions'

The ParameterDef type defines a parameter that can be referenced by name anywhere within an expression. Parameters are defined at the artifact level, and may be provided as part of the payload for an evaluation request. If no parameter value is provided, the default element is used to provide the value for the parameter. If no parameter or default is provided, the parameter is defined to be null.

Parameter Definition

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/31/2017. Last modified 11/1/2017

Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Parameter Definition to Element

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ default : Expression Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

◆ parameterTypeSpecifier : Type Specifier Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ accessLevel : Access Modifier Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

◆ parameterType : Public

[Is static False. Containment is Not Specified.]

ATTRIBUTES

Parameter Reference

Class in package 'Expressions'

The ParameterRef expression allows the value of a parameter to be referenced as part of an expression.

Parameter Reference
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Parameter Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ libraryName : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Ternary Expression

Class in package 'Expressions'

The TernaryExpression type defines the abstract base type for expressions that take three arguments.

Ternary Expression
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Ternary Expression to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public

Multiplicity: ([3], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Tuple Element Definition

Class in package 'Expressions'

TupleElementDefinition defines the name and type of a single element within a TupleTypeSpecifier.

Tuple Element Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Tuple Element Definition to Element

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ type : Type Specifier Public

[Is static False. Containment is Not Specified.]

Tuple Type Specifier

Class in package 'Expressions'

TupleTypeSpecifier defines the possible elements of a tuple.

Tuple Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Tuple Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ elements : Tuple Element Definition Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Type Specifier

Class in package 'Expressions'

TypeSpecifier is the abstract base type for all type specifiers.

Type Specifier
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017
 Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Type Specifier to Element

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

➡ Generalization from List Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

➡ Generalization from Named Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

➡ Generalization from Tuple Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

➡ Generalization from Interval Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

Unary Expression

Class in package 'Expressions'

The UnaryExpression type defines the abstract base type for expressions that take a single argument.

Unary Expression
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/31/2017. Last modified 11/1/2017
 Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Unary Expression to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public

[Is static False. Containment is Not Specified.]

External Data diagram

Class diagram in package 'Current KAS KNART'

External Data
Version 1.0

Iconstab created on 10/31/2017. Last modified 10/31/2017

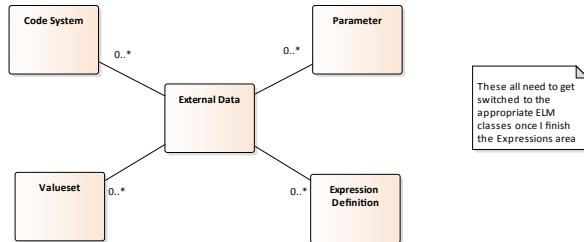


Figure 5: External Data

Note

Note in package 'Current KAS KNART'

These all need to get switched to the appropriate ELM classes once I finish the Expressions area

Note

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 10/31/2017
Extends

Code System

Class in package 'External Data'

A code system definition that can be referenced anywhere within the artifact.

Code System

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS
<p> Association (direction: Unspecified)</p> <p>Source: Public (Class) External Data Target: Public (Class) Code System Cardinality: [0..*]</p>

Expression Definition

Class in package 'External Data'

The named expression used to retrieve external data. For instance, an expression to retrieve patient demographic data or a set of SNOMED-CT codes subsumed by another SNOMED-CT code from a terminology server.

Expression Definition

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS

- ✓ Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Expression Definition

Cardinality: [0..*]

External Data

Class in package 'External Data'

External Data

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/31/2017. Last modified 10/31/2017

ASSOCIATIONS

- ✓ Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Valueset

Cardinality: [0..*]

- ✓ Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Expression Definition

Cardinality: [0..*]

- ✓ Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Parameter

Cardinality: [0..*]

- ✓ Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Code System

Cardinality: [0..*]

Parameter

Class in package 'External Data'

The parameter element define a parameters for the knowledge document. Parameters are expected to be provided by the caller when an evaluation is performed. Parameters can be referenced within any expression using a ParameterRef expression. For instance, one may define a MonthThreshold parameter, and use this parameter to determine whether the span of time that has elapsed since the last A1C has been performed warrants the elicitation of a reminder.

Parameter

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Parameter
Cardinality: [0..*]

Valueset

Class in package 'External Data'

A value set definition that can be referenced anywhere within the artifact.

Valueset

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Valueset
Cardinality: [0..*]

KnowledgeDocument diagram

Class diagram in package 'Current KAS KNART'

This file defines the root knowledge document type and element.

KnowledgeDocument

Version 1.0

Iconstab created on 10/18/2017. Last modified 2/23/2018

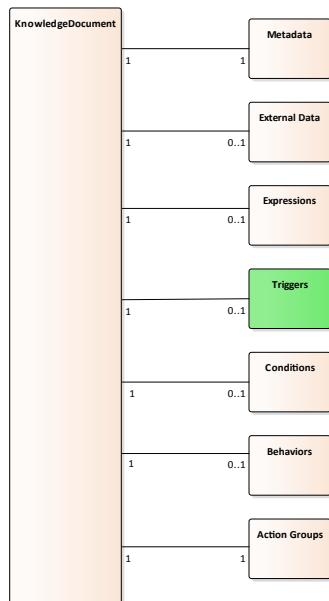


Figure 6: KnowledgeDocument

Behaviors

Class in package 'KnowledgeDocument'

The behaviors section defines the set of behaviors for this knowledge document. While there are no artifact-level behaviors defined at this time, this element is included as a point of extension, should it be needed.

Behaviors
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 10/31/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Behaviors Cardinality: [0..1]

Conditions

Class in package 'KnowledgeDocument'

The conditions section lists all conditions that pertain to the knowledge artifact. Conditions define the logic that determine the applicability of the artifact in the given context, any precondition or post condition, and/or any inclusion and exclusion criteria for the given CDS artifact. Conditions are structured as expressions to be evaluated in the target system.

Conditions
Version 1.0 Phase 1.0 Proposed
[constab created on 10/18/2017. Last modified 11/1/2017]

ASSOCIATIONS

ASSOCIATIONS	
Cardinality: [1]	Cardinality: [0..1]

Expressions

Class in package 'KnowledgeDocument'

The expressions section allows a CDS artifact author to define 'named expressions' that can be referenced anywhere within expressions in the artifact. This allows expression logic to be reused, as well as to be organized for readability and maintainability.

Expressions
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Expressions Cardinality: [0..1]

External Data

Class in package 'KnowledgeDocument'

The externalData section allows a CDS artifact author to define 'named expressions' to fetch information from an external source and bind this information to the 'context' of the knowledge artifact for later reference by the logic modules (e.g., the condition for the knowledge artifact or actions). It is the responsibility of the implementation to determine the nature of this boundary and how to fetch this information. For instance, one may write an expression to retrieve from a patient vMR the age of a patient or a list of clinical problems whose problem code are contained in a given ICD-9 value set. The age value and the list of patient problems may then be used in the 'condition' section of the same knowledge artifact to determine the applicability of the knowledge document to the given patient.

External Data
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) External Data Cardinality: [0..1]

KnowledgeDocument

Class in package 'KnowledgeDocument'

A knowledgeDocument represents a serialized Clinical Decision Support (CDS) knowledge artifact. It is intended to define a general serialization format for an Order Set, a Event-Condition-Action (ECA) Rule, and a Document Template. Given the general intended purpose for this schema, it is important to note that the serialization of a given CDS artifact is

defined by both this schema, in conjunction with a template defining artifact-specific constraints. For additional information on these artifact-specific constraints, please refer to the Implementation Guide.

KnowledgeDocument
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 11/29/2017

INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Generalization from ECA Rules to KnowledgeDocument
[Direction is 'Source -> Destination'.]
- ⇒ Generalization from Value Sets to KnowledgeDocument
[Direction is 'Source -> Destination'.]
- ⇒ Generalization from Documentation Template to KnowledgeDocument
[Direction is 'Source -> Destination'.]
- ⇒ Generalization from Business Processes to KnowledgeDocument
[Direction is 'Source -> Destination'.]
- ⇒ Generalization from Order Set to KnowledgeDocument
[Direction is 'Source -> Destination'.]

ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Expressions Cardinality: [0..1]
--	---
- ✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Conditions Cardinality: [0..1]
--	--
- ✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Metadata Cardinality: [1]
--	---
- ✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Groups Cardinality: [1]
--	--
- ✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Behaviors Cardinality: [0..1]
--	---

ASSOCIATIONS	
Association (direction: Unspecified) Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Group Cardinality: [1]
Association (direction: Unspecified) Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Triggers Cardinality: [0..1]
Association (direction: Unspecified) Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) External Data Cardinality: [0..1]

Action Groups

Class in package 'Metadata'

The actionGroups element is the top-level container for the groups of actions that make up a given knowledge document. This container defines the main content of the knowledge artifact: logical grouping constructs such as the clinical sections and orderables in an order set, the tasks to be performed by a rule, or the sections and menu choices that make up a document template.

Action Groups
Version 1.0 Phase 1.0 Proposed
lconstab created on 11/1/2017. Last modified 11/1/2017

ASSOCIATIONS	
Association (direction: Unspecified) Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Groups Cardinality: [1]

Metadata

Class in package 'Metadata'

The metadata section of the knowledge document defines the core metadata associated with this CDS knowledge artifact such as (1) the unique identifier for this artifact, (2) the unique identifier for its associated template(s), (3) the title and description of the artifact, (4) the status and history of the artifact, (5) any relevant entities associated with this artifact, and (6) information needed to categorize and retrieve the artifact.

The container for all of the metadata associated with a CDS knowledge artifact. Ideally, the metadata for artifacts is provided independently by the publisher for determining which artifact to retrieve.

Metadata
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ Identifiers : Public Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)</p> <p>Each version of a CDS knowledge artifact may have more than one identifier associated with it. Note that each identifier must be globally unique in the universe of CDS knowledge artifacts in which a particular artifact resides. [Is static False. Containment is Not Specified.]</p>
<p>◆ Artifact Type : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Currently three types of artifacts are in scope for Health eDecisions Use Case #1: order sets, event-condition-action rules, and documentation templates. Additional types will be added in future revisions of the standard. [Is static False. Containment is Not Specified.]</p>
<p>◆ SchemaIdentifier : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>This is the identifier of the XML schema (and its version) which governs the structure of this CDS Knowledge Artifact. [Is static False. Containment is Not Specified.]</p>
<p>◆ TemplateIds : int Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)</p> <p>These are the identifiers of templates which further constrain the structure of this knowledge artifact. [Is static False. Containment is Not Specified.]</p>
<p>◆ Title : Public</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Description : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>A description of the model. [Is static False. Containment is Not Specified.]</p>
<p>◆ KeyTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Provides a list of coded key terms that pertain to this artifact. [Is static False. Containment is Not Specified.]</p>
<p>◆ Categories : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Provides a list of coded categories to which this artifact belongs. [Is static False. Containment is Not Specified.]</p>
<p>◆ Language : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>[Is static False. Containment is Not Specified.]</p>

ATTRIBUTES	
◆ Status : Public	[Is static False. Containment is Not Specified.]
◆ UsageTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]
ASSOCIATIONS	
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Related Resources Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Publishers Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Data Models Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Libraries Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Supporting Evidence Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Contributions Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) EventHistory Cardinality: [0..1]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Usage Terms Cardinality: [0..*]

ASSOCIATIONS	
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p>	<p>Target: Public (Class) Applicability Cardinality: [0..1]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) KnowledgeDocument Cardinality: [1]</p>	<p>Target: Public (Class) Metadata Cardinality: [1]</p>

Triggers

Class in package 'Triggers'

Triggers
Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ASSOCIATIONS	
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) KnowledgeDocument Cardinality: [1]</p>	<p>Target: Public (Class) Triggers Cardinality: [0..1]</p>

MetaData diagram

Class diagram in package 'Current KAS KNART'

MetaData
Version 1.0
lconstab created on 10/18/2017. Last modified 10/30/2017

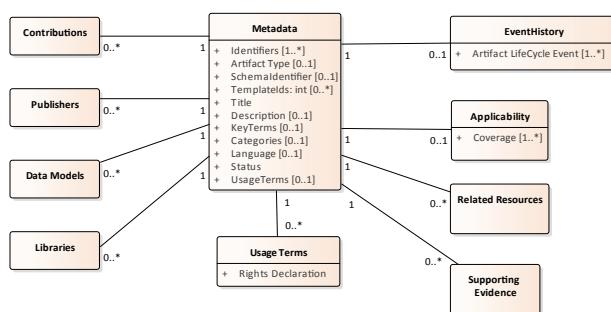


Figure 7: MetaData

Applicability

Class in package 'Metadata'

Specifies the conditions under which this artifact is applicable.

Applicability
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES

- ◆ Coverage : Public
Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)

Specifies various attributes of the patient population for whom and/or environment of care in which the CDS artifact is applicable.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Applicability
Cardinality: [0..1]

Contributions

Class in package 'Metadata'

A contribution is made by a specific contributor (organization, person, etc.), and was made in a particular way, as specified by the contributor's role. For example, a contributor may have been an author, or may have been a reviewer.

Includes a list of people and/or organizations who have contributed to the development of this artifact. Contributions are not necessarily tied to specific versions of the artifact.

Contributions
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

- ✍ Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Contributions
Cardinality: [0..*]

Data Models

Class in package 'Metadata'

Set of data models referenced in the Expression objects in this knowledge artifact.

Data Models
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Data Models
Cardinality: [0..*]

EventHistory

Class in package 'Metadata'

This is the history of events which have occurred for this particular version of the artifact.

EventHistory

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES

- Artifact LifeCycle Event : Public
Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)

An event in the life cycle of an artifact. Both the type of event are specified, as well as the point in time in which that event took place.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) EventHistory
Cardinality: [0..1]

Libraries

Class in package 'Metadata'

Set of libraries referenced by this artifact. Components of referenced libraries may be referenced by this artifact.

Libraries

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Libraries
Cardinality: [0..*]

Metadata

Class in package 'Metadata'

The metadata section of the knowledge document defines the core metadata associated with this CDS knowledge artifact such as (1) the unique identifier for this artifact, (2) the unique identifier for its associated template(s), (3) the title and description of the artifact, (4) the status and history of the artifact, (5) any relevant entities associated with this artifact, and (6) information needed to categorize and retrieve the artifact.

The container for all of the metadata associated with a CDS knowledge artifact. Ideally, the metadata for artifacts is provided independently by the publisher for determining which artifact to retrieve.

Metadata
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ Identifiers : Public Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)</p> <p>Each version of a CDS knowledge artifact may have more than one identifier associated with it. Note that each identifier must be globally unique in the universe of CDS knowledge artifacts in which a particular artifact resides. [Is static False. Containment is Not Specified.]</p>
<p>◆ Artifact Type : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Currently three types of artifacts are in scope for Health eDecisions Use Case #1: order sets, event-condition-action rules, and documentation templates. Additional types will be added in future revisions of the standard. [Is static False. Containment is Not Specified.]</p>
<p>◆ SchemaIdentifier : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>This is the identifier of the XML schema (and its version) which governs the structure of this CDS Knowledge Artifact. [Is static False. Containment is Not Specified.]</p>
<p>◆ TemplateIds : int Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)</p> <p>These are the identifiers of templates which further constrain the structure of this knowledge artifact. [Is static False. Containment is Not Specified.]</p>
<p>◆ Title : Public</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Description : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>A description of the model. [Is static False. Containment is Not Specified.]</p>
<p>◆ KeyTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p>

ATTRIBUTES	
Provides a list of coded key terms that pertain to this artifact.	[Is static False. Containment is Not Specified.]
◆ Categories : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	
Provides a list of coded categories to which this artifact belongs.	[Is static False. Containment is Not Specified.]
◆ Language : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]
◆ Status : Public	[Is static False. Containment is Not Specified.]
◆ UsageTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

ASSOCIATIONS	
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Related Resources Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Publishers Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Data Models Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Libraries Cardinality: [0..*]
✓ Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Supporting Evidence Cardinality: [0..*]
✓ Association (direction: Unspecified)	

ASSOCIATIONS	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Contributions Cardinality: [0..*]
Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) EventHistory Cardinality: [0..1]
Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Usage Terms Cardinality: [0..*]
Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Applicability Cardinality: [0..1]
Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Metadata Cardinality: [1]

Publishers

Class in package 'Metadata'

The set of people and/or organizations who publish the artifact.

Publishers
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Publishers Cardinality: [0..*]

Related Resources

Class in package 'Metadata'

A set of resources related to this artifact, along with an indication of the type of relationship. An artifact may be derived from or depend on other artifacts, along with other types of relationships. See the Artifact Lifecycle diagram in the Implementation Guide for more information.

Related Resources

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Related Resources
Cardinality: [0..*]

Supporting Evidence

Class in package 'Metadata'

The evidence grade and the sources of evidence associated with this artifact.

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Supporting Evidence
Cardinality: [0..*]

Usage Terms

Class in package 'Metadata'

This is the set of rights reserved by the person or organization holding the rights to this artifact, along with the set of permissions granted to consumers.

ATTRIBUTES

- Rights Declaration : Public

This specifies the intellectual property rights associated with this CDS knowledge artifact, including who the rights holder is and what rights they assert. It also specifies what permissions are granted for usage. The asserted rights and permissions are specified as a free-form text string.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Usage Terms
Cardinality: [0..*]

ASSOCIATIONS

Triggers diagram

Class diagram in package 'Current KAS KNART'

Triggers
Version 1.0
lconstab created on 2/23/2018. Last modified 2/23/2018



Figure 8: Triggers

eventType

Class «Enum» in package 'Triggers'

eventType
Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ATTRIBUTES
<p>◆ dataEvent : string Public An event in which a data item is created, removed, or accessed. Expression is expected to be an ExpressionRef that references an ExpressionDef in ExternalData that contains a Request with a triggerType attribute specified. [Is static False. Containment is Not Specified.]</p>
<p>◆ periodicEvent : string Public A time-based event which occurs at the specified period. Expression is expected to be a Period literal expression specifying the period on which the event should be repeated [Is static False. Containment is Not Specified.]</p>

Trigger

Class in package 'Triggers'

The triggers section defines the list of all triggers that 'activate' or 'trigger' the CDS knowledge artifact. For instance, opening a patient record may trigger a rule to execute if the conditions of the rule are met.

Trigger
Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ATTRIBUTES

ATTRIBUTES

- ◆ eventType : eventType Public

An enumeration of event types. Events occur external to the artifact that can be used as a trigger to the artifact.

[Is static False. Containment is Not Specified.]

Action Group

Package in package 'Current KAS KNART'

Action Group

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/30/2017. Last modified 10/30/2017

Action Base

Class in package 'Action Group'

Actions are the output of the CDS system and represent the tasks that must be carried out by a human or a computer system.

Action Base

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/18/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Generalization from Action Group to Action Base

[Direction is 'Source -> Destination'.]

- ⇒ Generalization from Atomic Action to Action Base

[Direction is 'Source -> Destination'.]

- ⇒ Generalization from Action Reference to Action Base

[Direction is 'Source -> Destination'.]

ATTRIBUTES

- ◆ actionId : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

An identifier for the action. The identifier must be unique within the scope of the artifact.

[Is static False. Containment is Not Specified.]

- ◆ number : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

A number for the action primarily for display to the user. This is different than the actionId which is used as a key to identify the action or group internally.

ATTRIBUTES	
	[Is static False. Containment is Not Specified.]
 order : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	
If specified, determines the ordering for this element within the group.	[Is static False. Containment is Not Specified.]
ASSOCIATIONS	
 Association (direction: Unspecified) Supporting Evidence	
The evidence grade and the sources of evidence associated with this artifact.	
Source: Public (Class) Action Base Cardinality: [1]	Target: Public (Class) Evidence Cardinality: [0..*]
 Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Actor Cardinality: [0..*]
 Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Behavior Cardinality: [0..*]
 Association (direction: Unspecified) Supporting Resources	
Didactic or other informational resources associated with the action that can be provided to the CDS recipient. Information resources can include inline text commentary and links to web resources.	
Source: Public (Class) Action Base Cardinality: [1]	Target: Public (Class) Knowledge Resource Cardinality: [0..*]
 Association (direction: Unspecified)	
Source: Public (Class) Action Base	Target: Public (Class) Condition Cardinality: [0..*]

Action Group

Class in package 'Action Group'

This type of action is used to organize a group of related actions into one container. The semantics of how the group's subelements interact with each other and how the subelements might be presented are specified in the group behavior.

Action Group
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 11/1/2017

Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Action Group to Action Base

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ title : Public

A brief title that is shown to the user of the artifact, i.e., the recipient of the CDS

[Is static False. Containment is Not Specified.]

◆ description : Public

A lengthier description of the action group that can be displayed to the user or the recipient of the CDS

[Is static False. Containment is Not Specified.]

◆ representedConcepts : Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

The concept(s) represented by this action group. For instance, a section may represent a group of beta-blockers, a composite orderable such as an insulin sliding scale, or a set of order sentences for a particular orderable (e.g., Tylenol).

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Class) Sub Elements «union»

Cardinality: [0..*]

Target: Public (Class) Action Group

✍ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument

Cardinality: [1]

Target: Public (Class) Action Group

Cardinality: [1]

Action Reference

Class in package 'Action Group'

This subelement specifies an action to be included by reference from a referenced library as defined in the libraries section of the metadata

A reference to an action defined in a library.

Action Reference
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Action Reference to Action Base	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ referencedActionId : Public	The Id of the action being referenced.
	[Is static False. Containment is Not Specified.]
◆ libraryName : Public	
	[Is static False. Containment is Not Specified.]

Actor

Class in package 'Action Group'

The actor that is responsible for executing the action. This element is used when an artifact can have multiple actors responsible for the execution of various CDS actions, for example, an interdisciplinary plan of care.

It is important to note the distinction between a CDS action and an ensuing clinical action. A CDS action might be to order patient ambulation, the actor for which is a physician responsible for writing the order. A nurse might be responsible for ensuring that the patient ambulates. In this case, the artifact will specify the physician as the actor.

Actor
Version 1.0 Phase 1.0 Proposed
lconstab created on 11/7/2017. Last modified 11/7/2017
Extends Party

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Actor to Party	[Direction is 'Source -> Destination'.]

ASSOCIATIONS	
✍ Association (direction: Unspecified)	

Source: Public (Class) Action Base

Target: Public (Class) Actor
Cardinality: [0..*]

Atomic Action

Class in package 'Action Group'

The subelement is an atomic or single action.

An action that is not further broken down into constituent actions.

Atomic Action

Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Action Base

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Atomic Action to Action Base

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Create Action to Atomic Action

[Direction is 'Source -> Destination'.]

⇒ Generalization from Remove Action to Atomic Action

[Direction is 'Source -> Destination'.]

⇒ Generalization from FireEventAction to Atomic Action

[Direction is 'Source -> Destination'.]

⇒ Generalization from Update Action to Atomic Action

[Direction is 'Source -> Destination'.]

⇒ Generalization from Declare Response Action to Atomic Action

[Direction is 'Source -> Destination'.]

⇒ Generalization from Collect Information Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ textEquivalent : Public

A brief textual description of the action that summarizes the action

[Is static False. Containment is Not Specified.]

Collect Information Action

Class in package 'Action Group'

This action requests information from the actor. The information request is specified as a DocumentationItem.

Collect Information Action
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 10/30/2017. Last modified 11/1/2017
 Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Collect Information Action to Atomic Action	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
documentationConcept : Public	This provides a specification of the information to be collected from the user. [Is static False. Containment is Not Specified.]
initialValue : Expression Public	An expression to compute an initial value for the documentation concept. The initial value could be computed from previous data about the patient available via expressions specified in the external data [Is static False. Containment is Not Specified.]
responseBinding : Response Binding Public	Defines the response binding for the documentation item. [Is static False. Containment is Not Specified.]

Create Action

Class in package 'Action Group'

A new action to be executed by a user or a computer system. The sentence provides the details of the action to be executed.

Create Action
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Create Action to Atomic Action	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
actionSentence : Expression Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The parameters of the action that is to be executed. For example, an action may be to order a medication. [Is static False. Containment is Not Specified.]

Declare Response Action

Class in package 'Action Group'

The DeclareResponseAction provides a mechanism to declare a container for responses provided by the user in response to CollectInformationActions. The intended semantics are to provide a container that can be used to access responses within expressions used subsequently in an artifact.

The DeclareResponseAction creates a named container within the Parameters scope of the artifact, and expressions may access the contents of a response using a ParameterRef expression. The container is expected to be a collection of name-value pairs, and the intended semantics are to allow the Property expression to be used, in connection with a ParameterRef expression as the source, to retrieve the current value for a property.

The CollectInformationAction contains a responseBinding attribute that specifies the name of the container, and the name of the property to be used to store the response value.

If no Name attribute is provided, the response container will be named Responses.

Declare Response Action
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Declare Response Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Evidence

Class in package 'Action Group'

Reference to research on which the artifact is based. This evidence can be 'graded' depending on its quality and pedigree and the strength of the recommendations it makes.

Evidence
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 10/30/2017

ATTRIBUTES

◆ qualityOfEvidence : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The quality of the evidence associated with this artifact. The codeSystem attribute specifies the quality scale used to grade this evidence source while the code specifies the actual quality score (represented as a coded value) associated with this evidence reference. CodeSystemName specifies the name of the scale. DisplayName specifies the display name of the coded value (the score).

[Is static False. Containment is Not Specified.]

◆ strengthOfRecommendation : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The strength of the recommendation assigned to this reference. The codeSystem attribute specifies the scale used to grade this

ATTRIBUTES

evidence source while the code specifies the actual score (represented as a coded value) for the strength of the evidence.
 CodeSystemName specifies the name of the scale. DisplayName specifies the display name of the coded value (the score).
 [Is static False. Containment is Not Specified.]

ASSOCIATIONS

Association (direction: Unspecified) Supporting Resource

Source: Public (Class) Evidence
 Cardinality: [1]

Target: Public (Class) Knowledge Resource
 Cardinality: [0..1]

Association (direction: Unspecified) Supporting Evidence

The evidence grade and the sources of evidence associated with this artifact.

Source: Public (Class) Action Base
 Cardinality: [1]

Target: Public (Class) Evidence
 Cardinality: [0..*]

FireEventAction

Class in package 'Action Group'

This action fires an event. The event can serve as a trigger to another artifact.

FireEventAction
 Version 1.0 Phase 1.0 Proposed
 lconstab created on 2/23/2018. Last modified 2/23/2018
 Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

Generalization from FireEventAction to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES

eventType : eventType Public

The type of the event that is fired.

[Is static False. Containment is Not Specified.]

actionSection : Expression Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

The action sentence is the payload of the event. Another artifact receives this payload as an input.

[Is static False. Containment is Not Specified.]

Knowledge Resource

Class in package 'Action Group'

KnowledgeResource specifies a reference to an associated resource of relevance to the artifact such as a guideline, a performance measure, another knowledge artifact, or a source of evidence for the artifact.

Knowledge Resource
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 10/30/2017

ATTRIBUTES	
◆ identifiers : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The set of unique identifiers for this resource. [Is static False. Containment is Not Specified.]
◆ templateIds : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The set of unique identifiers for the templates associated with this resource. [Is static False. Containment is Not Specified.]
◆ title : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The title of the document [Is static False. Containment is Not Specified.]
◆ location : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The URL of the given resource. [Is static False. Containment is Not Specified.]
◆ description : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	A short textual description of the resource. [Is static False. Containment is Not Specified.]
◆ citation : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	The resource citation. [Is static False. Containment is Not Specified.]

ASSOCIATIONS	
↙ Association (direction: Unspecified) Supporting Resource	Source: Public (Class) Evidence Cardinality: [1]
	Target: Public (Class) Knowledge Resource Cardinality: [0..1]

ASSOCIATIONS

Association (direction: Unspecified) Supporting Resources

Didactic or other informational resources associated with the action that can be provided to the CDS recipient. Information resources can include inline text commentary and links to web resources.

Source: Public (Class) Action Base

Cardinality: [1]

Target: Public (Class) Knowledge Resource

Cardinality: [0..*]

Party

Class in package 'Action Group'

Party

Version 1.0 Phase 1.0 Proposed
lconstab created on 11/7/2017. Last modified 11/7/2017

INCOMING STRUCTURAL RELATIONSHIPS

Generalization from Actor to Party

[Direction is 'Source -> Destination'.]

ATTRIBUTES

address : Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

An individual address of type AD associated with this entity.

[Is static False. Containment is Not Specified.]

contact : Public

Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

An individual contact item of type TEL associated with this entity.

[Is static False. Containment is Not Specified.]

Remove Action

Class in package 'Action Group'

This action removes another proposed action or an ongoing action.

Remove Action

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS

Generalization from Remove Action to Atomic Action

[Direction is 'Source -> Destination'.]

ATTRIBUTES
<p>◆ actionSentence : Expression Public The expression must resolve to the action that is being removed. [Is static False. Containment is Not Specified.]</p>

Response Binding

Class in package 'Action Group'

Defines the attributes required to specify a binding path for documentation item responses. The container attribute specifies the name of the response container that will be used. If no container attribute is provided, the default container name of Responses will be used. The property attribute specifies the name of the property within the container that will be used to store the user response value.

Response Binding
Version 1.0 Phase 1.0 Proposed
lconstab created on 11/1/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ container : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False) [Is static False. Containment is Not Specified.]</p>
<p>◆ property : Public [Is static False. Containment is Not Specified.]</p>

Update Action

Class in package 'Action Group'

This action changing the value of another existing action. The action being modified may be a proposed action (e.g., an medication being prescribed by a clinician) or be an ongoing action (e.g., an existing prescription). In these cases, a modification can be used to change the dose of the medication. It may also be used to discontinue a medication by changing the stop date.

Update Action
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 11/1/2017
Extends Atomic Action

OUTGOING STRUCTURAL RELATIONSHIPS
<p>↳ Generalization from Update Action to Atomic Action [Direction is 'Source -> Destination'.]</p>

ATTRIBUTES
<p>◆ actionSentence : Expression Public The modification to the action. This is specified by modifying the properties of an existing action using an ObjectRedefine</p>

ATTRIBUTES
expression. [Is static False. Containment is Not Specified.]

Behaviors

Package in package 'Current KAS KNART'

Behaviors

Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 10/31/2017

Behavior

Class in package 'Behaviors'

The behaviors associated with how the action is presented and executed. The semantics and the validity of behaviors for actions are described elsewhere.

A behavior may be specified for a specific action or a group of actions. This is the base type for all Behaviors.

Behavior

Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Group Selection Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Precheck Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Cardinality Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Required Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Visual Style Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Group Organization Behavior to Behavior
[Direction is 'Source -> Destination'.]

⇒ Generalization from Read Only Behavior to Behavior
[Direction is 'Source -> Destination'.]

ASSOCIATIONS

↗ Association (direction: Unspecified)

Source: Public (Class) Action Base

Target: Public (Class) Behavior
Cardinality: [0..*]

Cardinality Behavior

Class in package 'Behaviors'

For a group or an action, specifies if that item may be repeated.

Cardinality Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Cardinality Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Cardinality Behavior Type Public

[Is static False. Containment is Not Specified.]

Cardinality Behavior Type

Class «Enum» in package 'Behaviors'

Defines behavior for an action or a group for how many times that item may be repeated, i.e., cardinality. For example, if a user is documenting lesions, the lesion element may be repeated several times, once for each occurrence of a lesion on the patient or tissue sample or image

Cardinality Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

◆ Single : Public

An action with this behavior is one of the most frequent actions that is, or should be, included by an end user, for the particular context in which the action occurs. The system displaying the action to the end user should consider "pre-checking" such an action as a convenience for the user.

[Is static False. Containment is Not Specified.]

◆ Multiple : Public

An action with this behavior is one of the less frequent actions included by the end user, for the particular context in which the action occurs. The system displaying the actions to the end user would typically not "pre-check" such an action.

[Is static False. Containment is Not Specified.]

Group Organization Behavior

Class in package 'Behaviors'

For a group of actions, specifies the organizational intent of the grouping. This is meant to provide a hint to the system which displays the group of actions to an end user.

Group Organization Behavior
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Group Organization Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Group Organization Behavior Type Public

[Is static False. Containment is Not Specified.]

Group Organization Behavior Type

Class «Enum» in package 'Behaviors'

Defines organization behavior of a group: gives the reason why the items are grouped together.

Group Organization Behavior Type
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

◆ VisualGroup : Public

Any group marked with this behavior should be displayed as a visual group to the end user.

[Is static False. Containment is Not Specified.]

◆ LogicalGroup : Public

A group with this behavior logically groups its sub-elements, and may be shown as a visual group to the end user, but it is not required to do so.

[Is static False. Containment is Not Specified.]

◆ SentenceGroup : Public

A group of related alternative actions is a sentence group if the item referenced by the action is the same in all the actions, and each action simply constitutes a different variation on how to specify the details for that item. For example, two actions that could be in a SentenceGroup are "aspirin, 500 mg, 2 times per day" and "aspirin, 300 mg, 3 times per day". In both cases, aspirin is the item referenced by the action, and the two actions represent two different options for how aspirin might be ordered for the patient. Note that a SentenceGroup would almost always have an associated selection behavior of "AtMostOne", unless it's a required action, in which case, it would be "ExactlyOne".

[Is static False. Containment is Not Specified.]

Group Selection Behavior

Class in package 'Behaviors'

For a group of actions, specifies the number of actions that may be chosen by an end user.

Group Selection Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Group Selection Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Group Selection Behavior Type Public

[Is static False. Containment is Not Specified.]

Group Selection Behavior Type

Class «Enum» in package 'Behaviors'

Defines selection behavior of a group: specifies the number of selectable items in the group that may be selected by the end user when the items of the group are displayed.

Group Selection Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

◆ Any : Public

[Is static False. Containment is Not Specified.]

◆ All : Public

[Is static False. Containment is Not Specified.]

◆ AllOrNone : Public

[Is static False. Containment is Not Specified.]

◆ ExactlyOne : Public

[Is static False. Containment is Not Specified.]

◆ AtMostOne : Public

[Is static False. Containment is Not Specified.]

◆ OneOrMore : Public

[Is static False. Containment is Not Specified.]

Precheck Behavior

Class in package 'Behaviors'

For a particular action, specifies how often the action is expected to be selected in the particular context of the group containing that action. In general, depending on the group selection behavior, there may be zero, one or more actions which are frequently selected. This setting can serve as a hint to the system that displays the action to the end user: some systems will pre-select those actions which are (or should be) most frequently selected.

Precheck Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Precheck Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Precheck Behavior Type Public

[Is static False. Containment is Not Specified.]

Precheck Behavior Type

Class «Enum» in package 'Behaviors'

Defines selection frequency behavior for an action or group; i.e., for most frequently selected items, the end-user system may provide convenience options in the UI (such as pre-selection) in order to (1) communicate to the end user what the most frequently selected item is, or should, be in a particular context, and (2) save the end user time.

Precheck Behavior Type
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

◆ Yes : Public

An action with this behavior is one of the mos frequent actions that is, or should be, included by an end user, for the particular context in which the action occurs. The system displaying the action to the end user should consider "pre-checking" such an action as a convenience for the user.

[Is static False. Containment is Not Specified.]

◆ No : Public

An action with this behavior is one of the less frequent actions included by the end user, for the particular context in which the action occurs. The system displaying the actions to the end user would typically not "pre-check" such an action.

[Is static False. Containment is Not Specified.]

Read Only Behavior

Class in package 'Behaviors'

For a particular action or action group, specifies whether the elements are read only.

Read Only Behavior
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Read Only Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : boolean Public

[Is static False. Containment is Not Specified.]

Required Behavior

Class in package 'Behaviors'

For a single action, specifies what level of requiredness is associated with the action.

Required Behavior
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Required Behavior to Behavior

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ value : Required Behavior Type Public

[Is static False. Containment is Not Specified.]

Required Behavior Type

Class «Enum» in package 'Behaviors'

Defines requiredness behavior for selecting an action or an action group; i.e., whether the action or action group is required or optional.

Required Behavior Type
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES
<p>◆ Must : Public An action with this behavior must be included in the actions processed by the end user; the end user may not choose not to include this action. [Is static False. Containment is Not Specified.]</p>
<p>◆ Could : Public An action with this behavior may be included in the set of actions processed by the end user. [Is static False. Containment is Not Specified.]</p>
<p>◆ MustUnlessDocumented : Public An action with this behavior must be included in the set of actions processed by the end user, unless the end user provides documentation as to why the action was not included. [Is static False. Containment is Not Specified.]</p>

Visual Style Behavior

Class in package 'Behaviors'

For a group or an action, specifies the visual style for the action.

Visual Style Behavior
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Behavior

OUTGOING STRUCTURAL RELATIONSHIPS		
<p>◀ Generalization from Visual Style Behavior to Behavior [Direction is 'Source -> Destination'.]</p>		
<p>ATTRIBUTES</p> <table border="1"> <thead> <tr> <th>ATTRIBUTES</th></tr> </thead> <tbody> <tr> <td> <p>◆ value : string Public [Is static False. Containment is Not Specified.]</p> </td></tr> </tbody> </table>	ATTRIBUTES	<p>◆ value : string Public [Is static False. Containment is Not Specified.]</p>
ATTRIBUTES		
<p>◆ value : string Public [Is static False. Containment is Not Specified.]</p>		

Conditions

Package in package 'Current KAS KNART'

Conditions
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

Condition

Class in package 'Conditions'

The conditions section lists all conditions that pertain to the action. Conditions define the logic that determine the applicability of the action in the given context, any precondition or post condition, and/or any inclusion and exclusion criteria for the given action.

A condition specifies when a knowledge component is to be executed. For example, an ECA rule uses an ApplicableScenario condition to determine whether or not the action described by the artifact should be executed.

A collection of conditions that are used to define whether various aspects of the artifact, such as whether or not a particular action should be executed, or whether a particular order set item is applicable to a given patient

Condition
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

- ◆ Condition Role : Condition Role Type Public

The role determines when to evaluate the expression and how to proceed based on the expression results. Different artifact types use different types of conditions to control various aspects of the artifact. See the condition role type enumeration documentation for more discussion.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

- ✓ Association (direction: Unspecified)

Source: Public (Class) Action Base

Target: Public (Class) Condition

Cardinality: [0..*]

Condition Role Type

Class «Enum» in package 'Conditions'

The roles that a condition plays in the execution of a component. Currently, only one role type is defined. Additional role types may be defined in the future (e.g., inclusion criteria, exclusion criteria)

Condition Role Type
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES

ATTRIBUTES

- ◆ ApplicableScenario : Public

This role type specifies that a condition is used to determine whether or not a particular knowledge component should be executed. If the expression evaluates to true, then the component is executed.

[Is static False. Containment is Not Specified.]

Expressions

Package in package 'Current KAS KNART'

Expressions

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 10/31/2017

Access Modifier

Class «Enum» in package 'Expressions'

The AccessModifier type is used to specify the access level for the various definitions within a library such as parameters, expressions, and functions. If no access modifier is specified, public is assumed. Private definitions can only be accessed within the library in which they are defined.

Access Modifier

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

ATTRIBUTES	
◆ Public : Public	[Is static False. Containment is Not Specified.]
◆ Private : Public	[Is static False. Containment is Not Specified.]

Binary Expression

Class in package 'Expressions'

The BinaryExpression type defines the abstract base type for expressions that take two arguments.

Binary Expression

Version 1.0 Phase 1.0 Proposed

Iconstab created on 10/31/2017. Last modified 11/1/2017

Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS	
◀ Generalization from Binary Expression to Expression	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ operand : Expression Public Multiplicity: ([2], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

Element

Class in package 'Expressions'

The Element type defines the abstract base type for all library elements in ELM.

Element
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from TypeSpecifier to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from ParameterDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from Expression to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from TupleElementDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from ExpressionDefinition to Element	[Direction is 'Source -> Destination'.]
⇒ Generalization from OperandDefinition to Element	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ localID : Public	[Is static False. Containment is Not Specified.]
◆ annotations : Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	
The annotation element provides a mechanism for decorating expressions with application-specific information such as translation hints, visual designer information, or debug symbols.	[Is static False. Containment is Not Specified.]

Expression

Class in package 'Expressions'

The clinical (sub) scenario in which this action is applicable. This scenario is additive to any scenarios specified in the containers of this action such as action groups or the knowledge document. In other words, this scenario does not override the scenario constraints specified in the container elements.

The Expression type defines the abstract base type for all expressions used in the ELM expression language.

Expression

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Expression to Element

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Parameter Reference to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Ternary Expression to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Expression Reference to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Operand Reference to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Unary Expression to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Identifier Reference to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Nary Expression to Expression

[Direction is 'Source -> Destination'.]

⇒ Generalization from Binary Expression to Expression

[Direction is 'Source -> Destination'.]

Expression Definition

Class in package 'Expressions'

The named expression used to retrieve external data. For instance, an expression to retrieve patient demographic data or a set of SNOMED-CT codes subsumed by another SNOMED-CT code from a terminology server.

The ExpressionDef type defines an expression and an associated name that can be referenced by any expression in the artifact. The name must be unique within the artifact. The context attribute specifies the scope of the execution and is used by the environment to determine when and how to evaluate the expression.

Expression Definition
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Expression Definition to Element

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Function Definition to Expression Definition

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ context : Public

[Is static False. Containment is Not Specified.]

◆ accessLevel : Access Modifier Public

[Is static False. Containment is Not Specified.]

◆ expression : Expression Public

[Is static False. Containment is Not Specified.]

Expression Reference*Class in package 'Expressions'*

The ExpressionRef type defines an expression that references a previously defined NamedExpression. The result of evaluating an ExpressionReference is the result of evaluating the referenced NamedExpression.

Expression Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Expression Reference to Expression

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Generalization from Function Reference to Expression Reference

[Direction is 'Source -> Destination'.]

ATTRIBUTES

ATTRIBUTES	
◆ name : Public	[Is static False. Containment is Not Specified.]
◆ libraryName : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

Function Definition

Class in package 'Expressions'

The FunctionDef type defines a named function that can be invoked by any expression in the artifact. Function names must be unique within the artifact. Functions may take any number of operands.

Function Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression Definition

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Function Definition to Expression Definition	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ operand : Operand Definition Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

Function Reference

Class in package 'Expressions'

The FunctionRef type defines an expression that invokes a previously defined function. The result of evaluating each operand is passed to the function.

Function Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression Reference

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Function Reference to Expression Reference	[Direction is 'Source -> Destination'.]

ATTRIBUTES	

ATTRIBUTES

◆ operand : Expression Public

[Is static False. Containment is Not Specified.]

Identifier Reference

Class in package 'Expressions'

The IdentifierRef type defines an expression that references an unresolved identifier. An unresolved identifier reference generally indicates an error condition. The implementation is free to attempt to resolve the identifier, but is also justified in throwing an error at compile-time (or run-time for an interpretive system) when an identifier ref is encountered.

Identifier Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Identifier Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ libraryName : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Interval Type Specifier

Class in package 'Expressions'

IntervalTypeSpecifier defines an interval type by specifying the point type. Any type can serve as the point type for an interval, so long as it supports comparison operators, minimum and maximum value determination, as well as predecessor and successor functions.

Interval Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Interval Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

List Type Specifier

Class in package 'Expressions'

ListTypeSpecifier defines a list type by specifying the type of elements the list may contain.

List Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from List Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

Named Type Specifier

Class in package 'Expressions'

NamedTypeSpecifier defines a type identified by a name, such as Integer, String, Patient, or Encounter.

Named Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Named Type Specifier to Type Specifier

[Direction is 'Source -> Destination'.]

Nary Expression

Class in package 'Expressions'

The NaryExpression type defines an abstract base class for an expression that takes any number of arguments, including zero.

Nary Expression
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Nary Expression to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public
Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

ATTRIBUTES**Operand Definition***Class in package 'Expressions'*

The OperandDef type defines an operand to a function that can be referenced by name anywhere within the body of a function definition.

Operand Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Operand Definition to Element

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ operandType : Public

[Is static False. Containment is Not Specified.]

◆ operandTypeSpecifier : Type Specifier Public

[Is static False. Containment is Not Specified.]

Operand Reference*Class in package 'Expressions'*

The OperandRef expression allows the value of an operand to be referenced as part of an expression within the body of a function definition.

Operand Reference
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Operand Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

ATTRIBUTES
[Is static False. Containment is Not Specified.]

Parameter Definition

Class in package 'Expressions'

The ParameterDef type defines a parameter that can be referenced by name anywhere within an expression. Parameters are defined at the artifact level, and may be provided as part of the payload for an evaluation request. If no parameter value is provided, the default element is used to provide the value for the parameter. If no parameter or default is provided, the parameter is defined to be null.

Parameter Definition
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS
<p>↳ Generalization from Parameter Definition to Element [Direction is 'Source -> Destination'.]</p>

ATTRIBUTES
<p>◆ default : Expression Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False) [Is static False. Containment is Not Specified.]</p>
<p>◆ parameterTypeSpecifier : Type Specifier Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False) [Is static False. Containment is Not Specified.]</p>
<p>◆ name : Public [Is static False. Containment is Not Specified.]</p>
<p>◆ accessLevel : Access Modifier Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False) [Is static False. Containment is Not Specified.]</p>
<p>◆ parameterType : Public [Is static False. Containment is Not Specified.]</p>

Parameter Reference

Class in package 'Expressions'

The ParameterRef expression allows the value of a parameter to be referenced as part of an expression.

Parameter Reference
Version 1.0 Phase 1.0 Proposed

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Parameter Reference to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ name : Public

[Is static False. Containment is Not Specified.]

◆ libraryName : Public

Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Ternary Expression

Class in package 'Expressions'

The TernaryExpression type defines the abstract base type for expressions that take three arguments.

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Ternary Expression to Expression

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ operand : Expression Public

Multiplicity: ([3], Allow duplicates: 0, Is ordered: False)

[Is static False. Containment is Not Specified.]

Tuple Element Definition

Class in package 'Expressions'

TupleElementDefinition defines the name and type of a single element within a TupleTypeSpecifier.

OUTGOING STRUCTURAL RELATIONSHIPS

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Tuple Element Definition to Element	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ name : Public	[Is static False. Containment is Not Specified.]
◆ type : Type Specifier Public	[Is static False. Containment is Not Specified.]

Tuple Type Specifier

Class in package 'Expressions'

TupleTypeSpecifier defines the possible elements of a tuple.

Tuple Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Type Specifier

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Tuple Type Specifier to Type Specifier	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ elements : Tuple Element Definition Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)	[Is static False. Containment is Not Specified.]

Type Specifier

Class in package 'Expressions'

TypeSpecifier is the abstract base type for all type specifiers.

Type Specifier
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/31/2017. Last modified 11/1/2017
Extends Element

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Type Specifier to Element	[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from List Type Specifier to Type Specifier	[Direction is 'Source -> Destination'.]
⇒ Generalization from Named Type Specifier to Type Specifier	[Direction is 'Source -> Destination'.]
⇒ Generalization from Tuple Type Specifier to Type Specifier	[Direction is 'Source -> Destination'.]
⇒ Generalization from Interval Type Specifier to Type Specifier	[Direction is 'Source -> Destination'.]

Unary Expression

Class in package 'Expressions'

The UnaryExpression type defines the abstract base type for expressions that take a single argument.

Unary Expression
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017
Extends Expression

OUTGOING STRUCTURAL RELATIONSHIPS	
← Generalization from Unary Expression to Expression	[Direction is 'Source -> Destination'.]
ATTRIBUTES	
◆ operand : Expression Public	[Is static False. Containment is Not Specified.]

External Data

Package in package 'Current KAS KNART'

External Data
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 10/31/2017

Code System

Class in package 'External Data'

A code system definition that can be referenced anywhere within the artifact.

Code System
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS	
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Code System Cardinality: [0..*]

Expression Definition

Class in package 'External Data'

The named expression used to retrieve external data. For instance, an expression to retrieve patient demographic data or a set of SNOMED-CT codes subsumed by another SNOMED-CT code from a terminology server.

Expression Definition
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS	
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Expression Definition Cardinality: [0..*]

External Data

Class in package 'External Data'

External Data
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 10/31/2017

ASSOCIATIONS	
Association (direction: Unspecified)	

ASSOCIATIONS	
Source: Public (Class) External Data	Target: Public (Class) Valueset Cardinality: [0..*]
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Expression Definition Cardinality: [0..*]
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Parameter Cardinality: [0..*]
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Code System Cardinality: [0..*]

Parameter

Class in package 'External Data'

The parameter element define a parameters for the knowledge document. Parameters are expected to be provided by the caller when an evaluation is performed. Parameters can be referenced within any expression using a ParameterRef expression. For instance, one may define a MonthThreshold parameter, and use this parameter to determine whether the span of time that has elapsed since the last A1C has been performed warrants the elicitation of a reminder.

Parameter
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS	
Association (direction: Unspecified)	
Source: Public (Class) External Data	Target: Public (Class) Parameter Cardinality: [0..*]

Valueset

Class in package 'External Data'

A value set definition that can be referenced anywhere within the artifact.

Valueset
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/31/2017. Last modified 11/1/2017

ASSOCIATIONS	

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Class) External Data

Target: Public (Class) Valueset

Cardinality: [0..*]

KnowledgeDocument

Package in package 'Current KAS KNART'

A knowledge document is an instance of a CDS knowledge artifact such as a rule, an order set, or a documentation template.

KnowledgeDocument
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/30/2017. Last modified 11/1/2017

Behaviors

Class in package 'KnowledgeDocument'

The behaviors section defines the set of behaviors for this knowledge document. While there are no artifact-level behaviors defined at this time, this element is included as a point of extension, should it be needed.

Behaviors
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 10/31/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Behaviors Cardinality: [0..1]

Conditions

Class in package 'KnowledgeDocument'

The conditions section lists all conditions that pertain to the knowledge artifact. Conditions define the logic that determine the applicability of the artifact in the given context, any precondition or post condition, and/or any inclusion and exclusion criteria for the given CDS artifact. Conditions are structured as expressions to be evaluated in the target system.

Conditions
Version 1.0 Phase 1.0 Proposed
Iconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Conditions Cardinality: [0..1]

Expressions

Class in package 'KnowledgeDocument'

The expressions section allows a CDS artifact author to define 'named expressions' that can be referenced anywhere

within expressions in the artifact. This allows expression logic to be reused, as well as to be organized for readability and maintainability.

Expressions
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument
Cardinality: [1]

Target: Public (Class) Expressions
Cardinality: [0..1]

External Data

Class in package 'KnowledgeDocument'

The externalData section allows a CDS artifact author to define 'named expressions' to fetch information from an external source and bind this information to the 'context' of the knowledge artifact for later reference by the logic modules (e.g., the condition for the knowledge artifact or actions). It is the responsibility of the implementation to determine the nature of this boundary and how to fetch this information. For instance, one may write an expression to retrieve from a patient vMR the age of a patient or a list of clinical problems whose problem code are contained in a given ICD-9 value set. The age value and the list of patient problems may then be used in the 'condition' section of the same knowledge artifact to determine the applicability of the knowledge document to the given patient.

External Data
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument
Cardinality: [1]

Target: Public (Class) External Data
Cardinality: [0..1]

KnowledgeDocument

Class in package 'KnowledgeDocument'

A KnowledgeDocument represents a serialized Clinical Decision Support (CDS) knowledge artifact. It is intended to define a general serialization format for an Order Set, a Event-Condition-Action (ECA) Rule, and a Document Template. Given the general intended purpose for this schema, it is important to note that the serialization of a given CDS artifact is defined by both this schema, in conjunction with a template defining artifact-specific constraints. For additional information on these artifact-specific constraints, please refer to the Implementation Guide.

KnowledgeDocument
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/29/2017

INCOMING STRUCTURAL RELATIONSHIPS

INCOMING STRUCTURAL RELATIONSHIPS	
⇒ Generalization from ECA Rules to KnowledgeDocument	[Direction is 'Source -> Destination'.]
⇒ Generalization from Value Sets to KnowledgeDocument	[Direction is 'Source -> Destination'.]
⇒ Generalization from Documentation Template to KnowledgeDocument	[Direction is 'Source -> Destination'.]
⇒ Generalization from Business Processes to KnowledgeDocument	[Direction is 'Source -> Destination'.]
⇒ Generalization from Order Set to KnowledgeDocument	[Direction is 'Source -> Destination'.]

ASSOCIATIONS	
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Expressions Cardinality: [0..1]
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Conditions Cardinality: [0..1]
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Metadata Cardinality: [1]
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Groups Cardinality: [1]
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Behaviors Cardinality: [0..1]
✍ Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Action Group Cardinality: [1]

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument	Target: Public (Class) Triggers
Cardinality: [1]	Cardinality: [0..1]
 Association (direction: Unspecified)	
Source: Public (Class) KnowledgeDocument	Target: Public (Class) External Data
Cardinality: [1]	Cardinality: [0..1]

Metadata

Package in package 'Current KAS KNART'

Metadata

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 10/30/2017

Action Groups

Class in package 'Metadata'

The actionGroups element is the top-level container for the groups of actions that make up a given knowledge document. This container defines the main content of the knowledge artifact: logical grouping constructs such as the clinical sections and orderables in an order set, the tasks to be performed by a rule, or the sections and menu choices that make up a document template.

Action Groups

Version 1.0 Phase 1.0 Proposed
lconstab created on 11/1/2017. Last modified 11/1/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument
Cardinality: [1]

Target: Public (Class) Action Groups
Cardinality: [1]

Applicability

Class in package 'Metadata'

Specifies the conditions under which this artifact is applicable.

Applicability

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES

Coverage : Public
Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)

Specifies various attributes of the patient population for whom and/or environment of care in which the CDS artifact is applicable.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Applicability
Cardinality: [0..1]

Contributions

Class in package 'Metadata'

A contribution is made by a specific contributor (organization, person, etc.), and was made in a particular way, as specified by the contributor's role. For example, a contributor may have been an author, or may have been a reviewer.

Includes a list of people and/or organizations who have contributed to the development of this artifact. Contributions are not necessarily tied to specific versions of the artifact.

Contributions
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Contributions
Cardinality: [0..*]

Data Models

Class in package 'Metadata'

Set of data models referenced in the Expression objects in this knowledge artifact.

Data Models
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Data Models
Cardinality: [0..*]

EventHistory

Class in package 'Metadata'

This is the history of events which have occurred for this particular version of the artifact.

EventHistory
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES

Artifact LifeCycle Event : Public
Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)

ATTRIBUTES

An event in the life cycle of an artifact. Both the type of event are specified, as well as the point in time in which that event took place.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) EventHistory
Cardinality: [0..1]

Libraries

Class in package 'Metadata'

Set of libraries referenced by this artifact. Components of referenced libraries may be referenced by this artifact.

Libraries

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]

Target: Public (Class) Libraries
Cardinality: [0..*]

Metadata

Class in package 'Metadata'

The metadata section of the knowledge document defines the core metadata associated with this CDS knowledge artifact such as (1) the unique identifier for this artifact, (2) the unique identifier for its associated template(s), (3) the title and description of the artifact, (4) the status and history of the artifact, (5) any relevant entities associated with this artifact, and (6) information needed to categorize and retrieve the artifact.

The container for all of the metadata associated with a CDS knowledge artifact. Ideally, the metadata for artifacts is provided independently by the publisher for determining which artifact to retrieve.

Metadata

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ATTRIBUTES

Identifiers : Public

Multiplicity: ([1..*], Allow duplicates: 0, Is ordered: False)

Each version of a CDS knowledge artifact may have more than one identifier associated with it. Note that each identifier must be globally unique in the universe of CDS knowledge artifacts in which a particular artifact resides.

[Is static False. Containment is Not Specified.]

ATTRIBUTES
<p>◆ Artifact Type : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Currently three types of artifacts are in scope for Health eDecisions Use Case #1: order sets, event-condition-action rules, and documentation templates. Additional types will be added in future revisions of the standard.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ SchemaIdentifier : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>This is the identifier of the XML schema (and its version) which governs the structure of this CDS Knowledge Artifact.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ TemplateIds : int Public Multiplicity: ([0..*], Allow duplicates: 0, Is ordered: False)</p> <p>These are the identifiers of templates which further constrain the structure of this knowledge artifact.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Title : Public</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Description : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>A description of the model.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ KeyTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Provides a list of coded key terms that pertain to this artifact.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Categories : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>Provides a list of coded categories to which this artifact belongs.</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Language : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ Status : Public</p> <p>[Is static False. Containment is Not Specified.]</p>
<p>◆ UsageTerms : Public Multiplicity: ([0..1], Allow duplicates: 0, Is ordered: False)</p> <p>[Is static False. Containment is Not Specified.]</p>

ATTRIBUTES
ASSOCIATIONS
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Related Resources Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Publishers Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Data Models Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Libraries Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Supporting Evidence Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Contributions Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) EventHistory Cardinality: [0..1]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Usage Terms Cardinality: [0..*]</p>
<p>✓ Association (direction: Unspecified)</p> <p>Source: Public (Class) Metadata Cardinality: [1]</p> <p>Target: Public (Class) Applicability Cardinality: [0..1]</p>
<p>✓ Association (direction: Unspecified)</p>

ASSOCIATIONS	
Source: Public (Class) KnowledgeDocument Cardinality: [1]	Target: Public (Class) Metadata Cardinality: [1]

Publishers

Class in package 'Metadata'

The set of people and/or organizations who publish the artifact.

Publishers
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Publishers Cardinality: [0..*]

Related Resources

Class in package 'Metadata'

A set of resources related to this artifact, along with an indication of the type of relationship. An artifact may be derived from or depend on other artifacts, along with other types of relationships. See the Artifact Lifecycle diagram in the Implementation Guide for more information.

Related Resources
Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Class) Metadata Cardinality: [1]	Target: Public (Class) Related Resources Cardinality: [0..*]

Sub Elements

Class «union» in package 'Metadata'

The constituent elements of the group can be of different types including subgroups, simple or atomic actions, and subgroups embedded by reference. While the group allows artifacts of different types to be mixed and matched in any combination, a particular type of artifact might further restrict the combinations.

For example, an artifact type might require subelements of a particular group to be either groups or simple actions; elements of both types cannot exist in the group.

Sub Elements

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/30/2017. Last modified 11/1/2017**ATTRIBUTES**

◆ simpleAction : Atomic Action Public

[Is static False. Containment is Not Specified.]

◆ actionGroup : Action Group Public

[Is static False. Containment is Not Specified.]

◆ actionRef : Action Reference Public

[Is static False. Containment is Not Specified.]

ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Class) Sub Elements «union»
Cardinality: [0..*]

Target: Public (Class) Action Group

Supporting Evidence*Class in package 'Metadata'*

The evidence grade and the sources of evidence associated with this artifact.

Supporting Evidence

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017**ASSOCIATIONS**

✍ Association (direction: Unspecified)

Source: Public (Class) Metadata
Cardinality: [1]Target: Public (Class) Supporting Evidence
Cardinality: [0..*]**Usage Terms***Class in package 'Metadata'*

This is the set of rights reserved by the person or organization holding the rights to this artifact, along with the set of permissions granted to consumers.

Usage Terms

Version 1.0 Phase 1.0 Proposed
lconstab created on 10/18/2017. Last modified 11/1/2017**ATTRIBUTES**

ATTRIBUTES Rights Declaration : Public

This specifies the intellectual property rights associated with this CDS knowledge artifact, including who the rights holder is and what rights they assert. It also specifies what permissions are granted for usage. The asserted rights and permissions are specified as a free-form text string.

[Is static False. Containment is Not Specified.]

ASSOCIATIONS Association (direction: Unspecified)

Source: Public (Class) Metadata

Cardinality: [1]

Target: Public (Class) Usage Terms

Cardinality: [0..*]

Triggers

Package in package 'Current KAS KNART'

Triggers

Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

Trigger

Class in package 'Triggers'

The triggers section defines the list of all triggers that 'activate' or 'trigger' the CDS knowledge artifact. For instance, opening a patient record may trigger a rule to execute if the conditions of the rule are met.

Trigger

Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ATTRIBUTES

◆ eventType : eventType Public

An enumeration of event types. Events occur external to the artifact that can be used as a trigger to the artifact.

[Is static False. Containment is Not Specified.]

Triggers

Class in package 'Triggers'

Triggers

Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ASSOCIATIONS

◆ Association (direction: Unspecified)

Source: Public (Class) KnowledgeDocument
Cardinality: [1]

Target: Public (Class) Triggers
Cardinality: [0..1]

eventType

Class «Enum» in package 'Triggers'

eventType

Version 1.0 Phase 1.0 Proposed
lconstab created on 2/23/2018. Last modified 2/23/2018

ATTRIBUTES

◆ dataEvent : string Public

An event in which a data item is created, removed, updated, or accessed. Expression is expected to be an ExpressionRef that

ATTRIBUTES
references an ExpressionDef in ExternalData that contains a Request with a triggerType attribute specified. [Is static False. Containment is Not Specified.]
 periodicEvent : string Public A time-based event which occurs at the specified period. Expression is expected to be a Period literal expression specifying the period on which the event should be repeated [Is static False. Containment is Not Specified.]