Current

Action Groups diagram

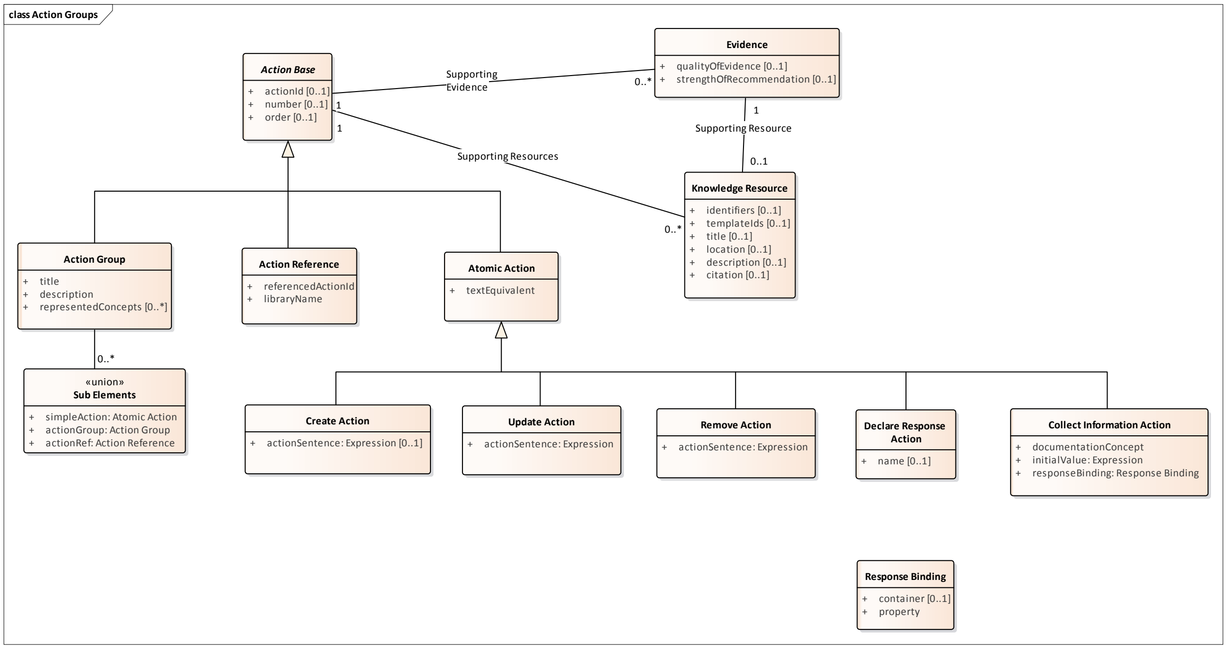

Figure 1: Action Groups

Name	**Description**
Action Base	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 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 which each other and how the subelements might be presented are specified in the group behavior.
Action Reference	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.
Atomic Action	The subelement is an atomic or single action. An action that is not further broken down into constituent actions.
Collect Information Action	This action requests information from the actor. The information request is specified as a DocumentationItem.
Create Action	A new action to be executed by a user or a computer system. The sentence provides the details of the action to be executed.
Declare Response Action	The DeclareResponseAction provides a mechanism to declare a container for responses provided by the user in response to

Name	Description
	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.
Evidence	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.
Knowledge Resource	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.
Remove Action	This action removes another proposed action or an ongoing action.
Response Binding	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.
Update Action	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.
Sub Elements	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.

Behaviors diagram

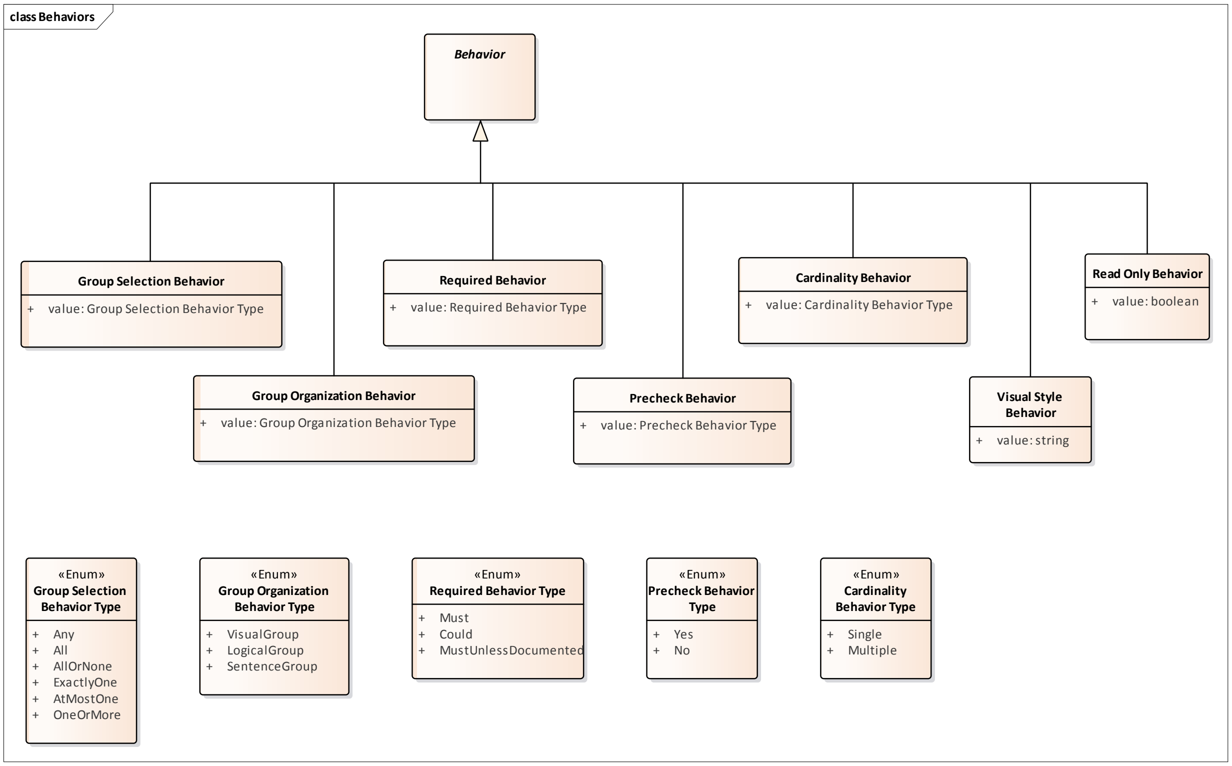
class Behaviors
Behavior
Group Selection Behavior
+ value: Group Selection Behavior Type
Required Behavior
+ value: Required Behavior Type
Cardinality Behavior
+ value: Cardinality Behavior Type
Read Only Behavior
+ value: boolean
Group Organization Behavior
+ value: Group Organization Behavior Type
Precheck Behavior
+ value: Precheck Behavior Type
Visual Style Behavior
+ value: string
«Enum» Group Selection Behavior Type
+ Any
+ All
+ AllOrNone
+ ExactlyOne
+ AtMostOne
+ OneOrMore
«Enum» Group Organization Behavior Type
+ VisualGroup
+ LogicalGroup
+ SentenceGroup
«Enum» Required Behavior Type
+ Must
+ Could
+ MustUnlessDocumented
«Enum» Precheck Behavior Type
+ Yes
+ No
«Enum» Cardinality Behavior Type
+ Single
+ Multiple

Figure 2: Behaviors

Name	**Description**
Behavior	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.
Cardinality Behavior	For a group or an action, specifies if that item may be repeated.
Cardinality Behavior Type	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
Group Organization Behavior	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 Type	Defines organization behavior of a group: gives the reason why the items are grouped together.
Group Selection Behavior	For a group of actions, specifies the number of actions that may be chosen by an end user.
Group Selection Behavior Type	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.

Name	Description
Precheck Behavior	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 Type	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.
Read Only Behavior	For a particular action or action group, specifies whether the elements are read only.
Required Behavior	For a single action, specifies what level of requiredness is associated with the action.
Required Behavior Type	Defines requiredness behavior for selecting an action or an action group; i.e., whether the action or action group is required or optional.
Visual Style Behavior	For a group or an action, specifies the visual style for the action.

Conditions diagram

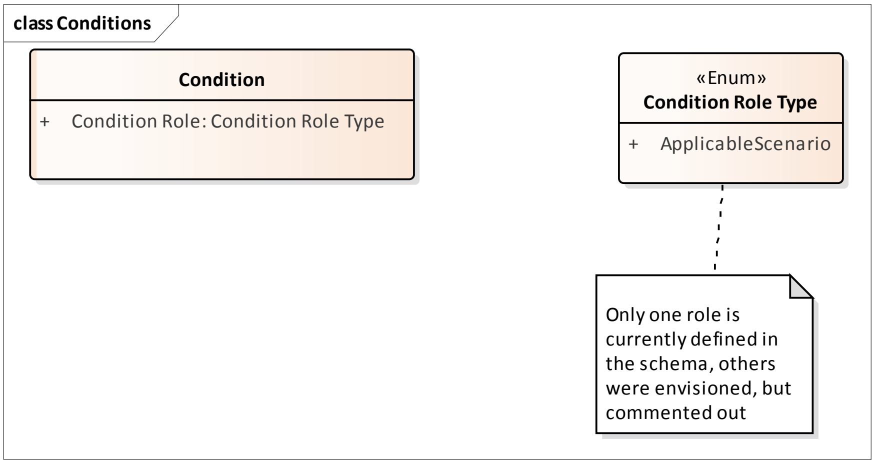
class Conditions
Condition
+ Condition Role: Condition Role Type
«Enum»
Condition Role Type
+ ApplicableScenario
Only one role is currently defined in the schema, others were envisioned, but commented out

Figure 3: Conditions

Name	Description
Note	Only one role is currently defined in the schema, others were envisioned, but commented out
Condition	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.

Name	Description
	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 Role Type	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)

Expressions diagram

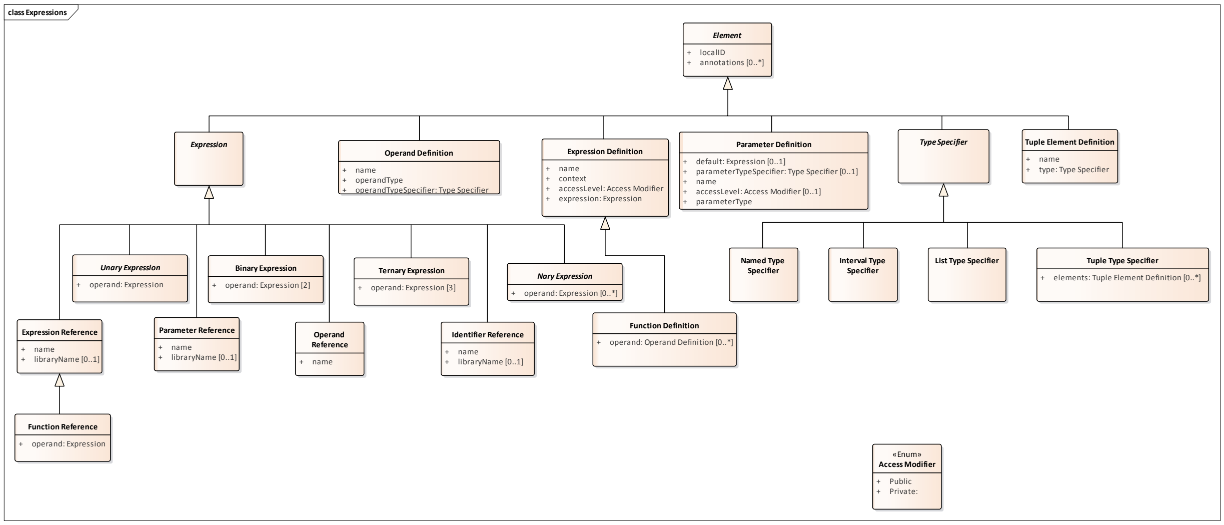

Figure 4: Expressions

Name	Description
Access Modifier	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.
Binary Expression	The BinaryExpression type defines the abstract base type for expressions that take two arguments.
Element	The Element type defines the abstract base type for all library elements in ELM.
Expression	The clinical (sub) scenario in which this action is applicable. This scenario is additive to any scenarios specified in the

Name	Description
	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 Definition	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 Reference	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.
Function Definition	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 Reference	The FunctionRef type defines an expression that invokes a previously defined function. The result of evaluating each operand is passed to the function.
Identifier Reference	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.
Interval Type Specifier	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.
List Type Specifier	ListTypeSpecifier defines a list type by specifying the type of elements the list may contain.
Named Type Specifier	NamedTypeSpecifier defines a type identified by a name, such as Integer, String, Patient, or Encounter.
Nary Expression	The NaryExpression type defines an abstract base class for an expression that takes any number of arguments, including zero.
Operand Definition	The OperandDef type defines an operand to a function that can be referenced by name anywhere within the body of a function definition.
Operand Reference	The OperandRef expression allows the value of an operand to be

Name	Description
	referenced as part of an expression within the body of a function definition.
Parameter Definition	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 Reference	The ParameterRef expression allows the value of a parameter to be referenced as part of an expression.
Ternary Expression	The TernaryExpression type defines the abstract base type for expressions that take three arguments.
Tuple Element Definition	TupleElementDefinition defines the name and type of a single element within a TupleTypeSpecifier.
Tuple Type Specifier	TupleTypeSpecifier defines the possible elements of a tuple.
Type Specifier	TypeSpecifier is the abstract base type for all type specifiers.
Unary Expression	The UnaryExpression type defines the abstract base type for expressions that take a single argument.

External Data diagram

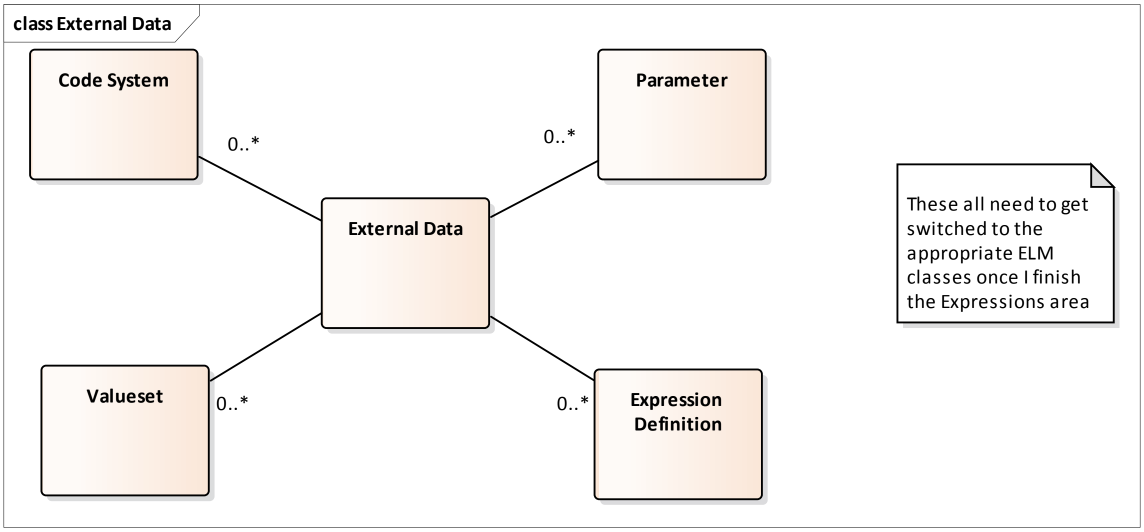

Figure 5: External Data

Name	Description
Note	These all need to get switched to the appropriate ELM classes once I finish the Expressions area
Code System	A code system definition that can be referenced anywhere within the artifact.
Expression Definition	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.

Name	Description
External Data	
Parameter	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.
Valueset	A value set definition that can be referenced anywhere within the artifact.

KnowledgeDocument diagram

This file defines the root knowledge document type and element.

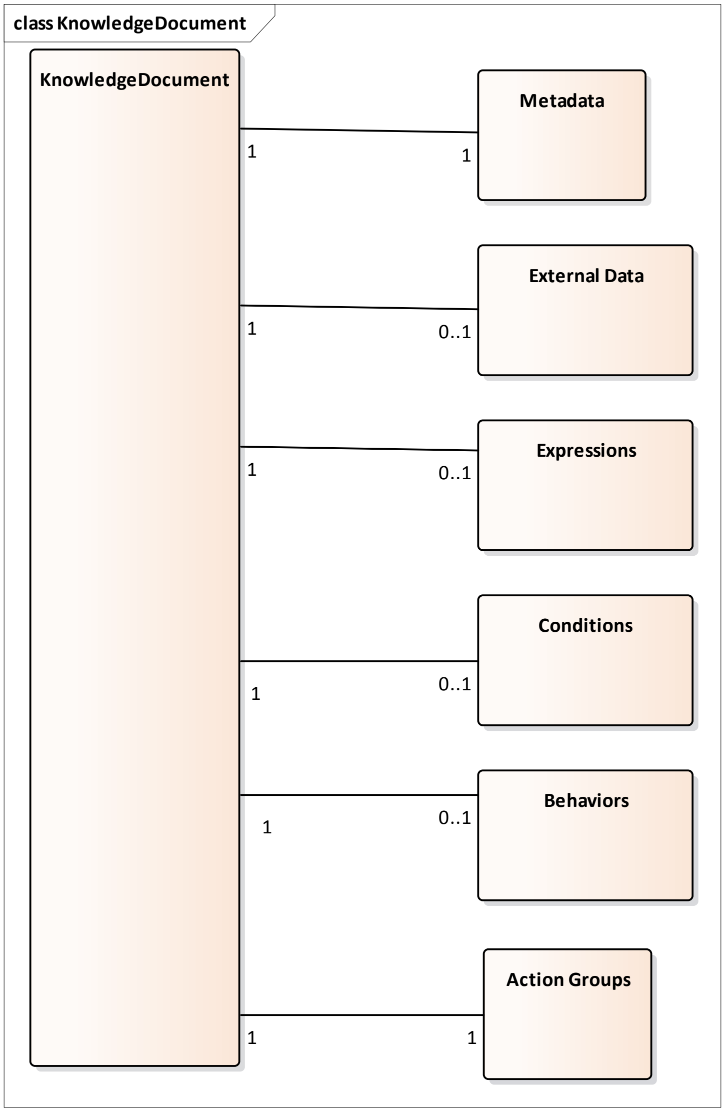

Figure 6: KnowledgeDocument

Name	Description
Behaviors	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.
Conditions	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.
Expressions	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.
External Data	**T**he 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.
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.
Action Groups	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.
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

Name	Description
	(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 diagram

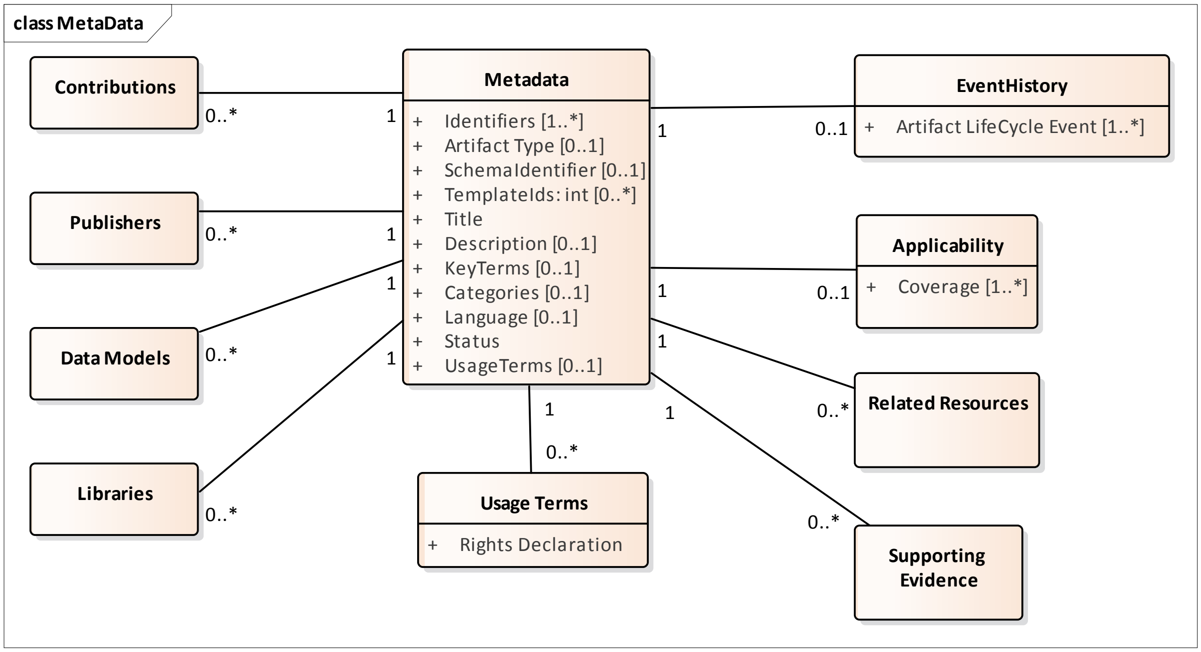

Figure 7: MetaData

Name	Description
Applicability	Specifies the conditions under which this artifact is applicable.
Contributions	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.
Data Models	Set of data models referenced in the Expression objects in this knowledge artifact.
EventHistory	This is the history of events which have occurred for this particular version of the artifact.
Libraries	Set of libraries referenced by this artifact. Components of referenced libraries may be referenced by this artifact.
Metadata	The metadata section of the knowledge document defines the

Name	Description
	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.
Publishers	The set of people and/or organizations who publish the artifact.
Related Resources	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.
Supporting Evidence	The evidence grade and the sources of evidence associated with this artifact.
Usage Terms	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.

Action Group

Name	Description
Action Base	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 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 which each other and how the subelements might be presented are specified in the group behavior.
Action Reference	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.
Atomic Action	The subelement is an atomic or single action. An action that is not further broken down into constituent actions.
Collect Information Action	This action requests information from the actor. The information request is specified as a DocumentationItem.
Create Action	A new action to be executed by a user or a computer system. The sentence provides the details of the action to be executed.
Declare Response Action	The DeclareResponseAction provides a mechanism to declare a

Name	Description
	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.
Evidence	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.
Knowledge Resource	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.
Remove Action	This action removes another proposed action or an ongoing action.
Response Binding	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.
Update Action	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.

Behaviors

Name	Description
Behavior	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.
Cardinality Behavior	For a group or an action, specifies if that item may be repeated.
Cardinality Behavior Type	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
Group Organization Behavior	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 Type	Defines organization behavior of a group: gives the reason why the items are grouped together.
Group Selection Behavior	For a group of actions, specifies the number of actions that may be chosen by an end user.
Group Selection Behavior Type	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.
Precheck Behavior	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 Type	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.
Read Only Behavior	For a particular action or action group, specifies whether the elements are read only.
Required Behavior	For a single action, specifies what level of requiredness is associated with the action.
Required Behavior Type	Defines requiredness behavior for selecting an action or an action group; i.e., whether the action or action group is required or optional.
Visual Style Behavior	For a group or an action, specifies the visual style for the action.

Conditions

Name	Description
Condition	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 Role Type	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)

Expressions

Name	Description
Access Modifier	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.
Binary Expression	The BinaryExpression type defines the abstract base type for expressions that take two arguments.
Element	The Element type defines the abstract base type for all library elements in ELM.
Expression	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 Definition	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 Reference	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.
Function Definition	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 Reference	The FunctionRef type defines an expression that invokes a previously defined function. The result of evaluating each operand is passed to the function.
Identifier Reference	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.
Interval Type Specifier	IntervalTypeSpecifier defines an interval type by specifying the

Name	Description
	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.
List Type Specifier	ListTypeSpecifier defines a list type by specifying the type of elements the list may contain.
Named Type Specifier	NamedTypeSpecifier defines a type identified by a name, such as Integer, String, Patient, or Encounter.
Nary Expression	The NaryExpression type defines an abstract base class for an expression that takes any number of arguments, including zero.
Operand Definition	The OperandDef type defines an operand to a function that can be referenced by name anywhere within the body of a function definition.
Operand Reference	The OperandRef expression allows the value of an operand to be referenced as part of an expression within the body of a function definition.
Parameter Definition	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 Reference	The ParameterRef expression allows the value of a parameter to be referenced as part of an expression.
Ternary Expression	The TernaryExpression type defines the abstract base type for expressions that take three arguments.
Tuple Element Definition	TupleElementDefinition defines the name and type of a single element within a TupleTypeSpecifier.
Tuple Type Specifier	TupleTypeSpecifier defines the possible elements of a tuple.
Type Specifier	TypeSpecifier is the abstract base type for all type specifiers.
Unary Expression	The UnaryExpression type defines the abstract base type for expressions that take a single argument.

External Data

Name	Description
Code System	A code system definition that can be referenced anywhere within the artifact.
Expression Definition	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.
External Data	
Parameter	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.
Valueset	A value set definition that can be referenced anywhere within the artifact.

KnowledgeDocument

Name	Description
Behaviors	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.
Conditions	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.
Expressions	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.
External Data	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.
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.

Metadata

Name	Description
Action Groups	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.
Applicability	Specifies the conditions under which this artifact is applicable.
Contributions	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.
Data Models	Set of data models referenced in the Expression objects in this knowledge artifact.
EventHistory	This is the history of events which have occurred for this particular version of the artifact.
Libraries	Set of libraries referenced by this artifact. Components of referenced libraries may be referenced by this artifact.
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.
Publishers	The set of people and/or organizations who publish the artifact.
Related Resources	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.
Sub Elements	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.

Name	Description
	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.
Supporting Evidence	The evidence grade and the sources of evidence associated with this artifact.
Usage Terms	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.