STF 476: TDL Phase 2

Status Report

Document History

- 2014-07-07: Document submitted for SG #3

- 2014-05-10: Document submitted for MTS #62

  - long form for SG #3 / Technical Session

  - short form for MTS #62

- 2014-03-19: Document submitted for SG #2

Task 0: Session Overview

- WK09 Feb 24-28 - Session 1 @ETSI

- WK15 Apr 07-11 - Session 2 @ETSI

- WK23 Jun 02-06 - Session 3 @FOKUS

- WK36 Sep 01-05 - Session 4 @ETSI

- WK42 Oct 13-17 - Session 5 @Siemens

- WK49 Dec 01-05 - Session 6 @ETSI

- Sessions 3-6 upgraded to 5 days!

Task 0: Milestone 2 Timeline

- WK23 Jun 02-06 - Session 3 @FOKUS

  - 5 experts, 25 days, review feedback, define roadmaps, prep. stable drafts

- WK36 Sep 01-05 - Session 4 @ETSI

  - 5 experts, 25 days, finalise stable drafts

- WK37 Sep 08-12 - Deliverables ready

- WK38 Sep 16-18 - UCAAT 2014

- WK40 Oct 01-02 - MTS #63 @Berlin

Session 3 Summary

- Goals

    - review feedback from Milestone 1

    - define roadmaps for Milestone 2

    - have technical discussions on data concepts and graphical syntax

    - lay down foundations for exchange format

- Notes

    - technical issues with the mailing list interfered with the organisation
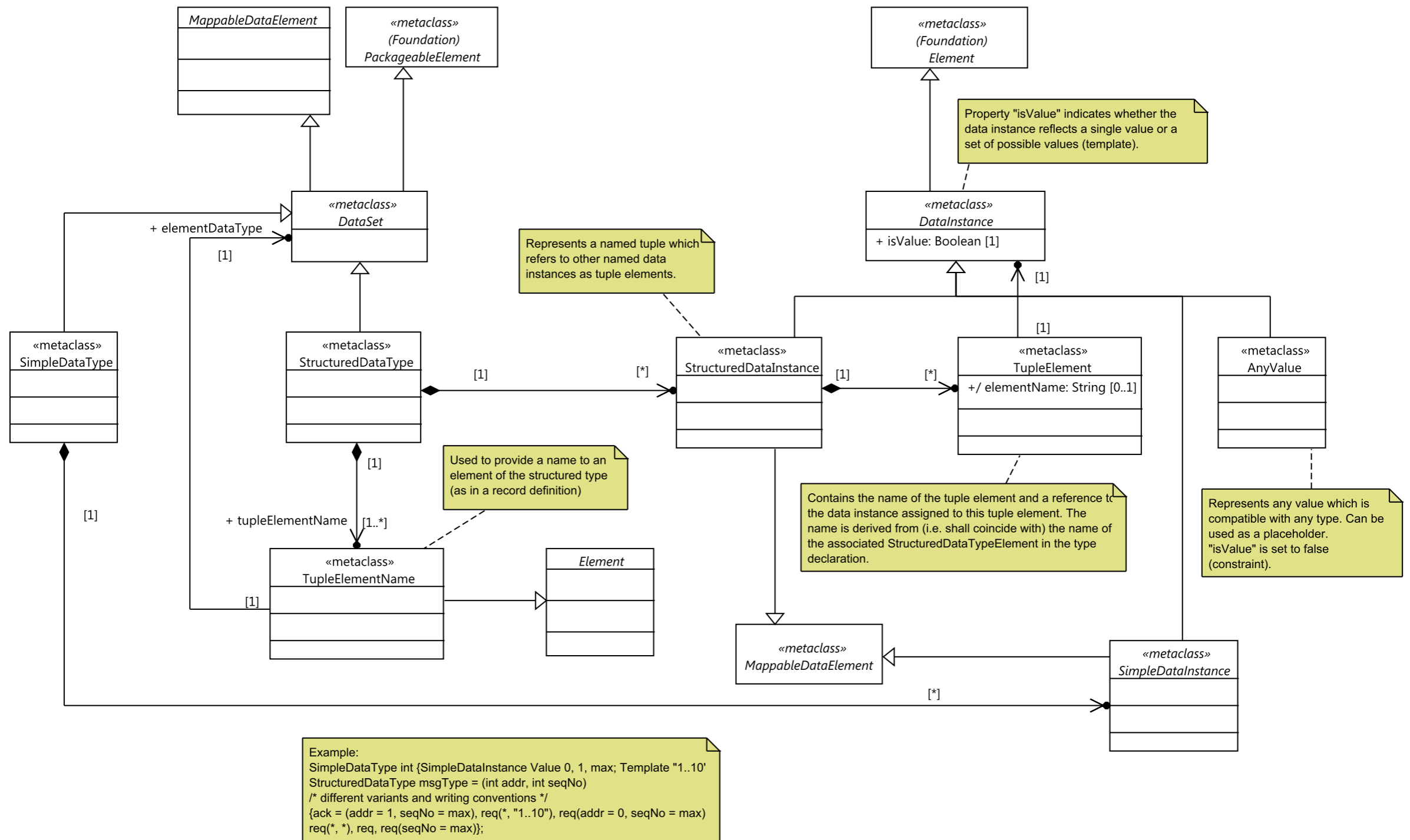
Feedback and Discussions

- Changing role assignments in test descriptions

  - different ways to achieve, postponed

  - are test architectures as a generic kind of test configurations desired?

- Different notions of SUT (SUT, IUT, OUT, … xUT)

  - concrete syntax can be used to refine these, no semantical impact

- Graphical concrete syntax

  - document structure independent from meta-model document structure
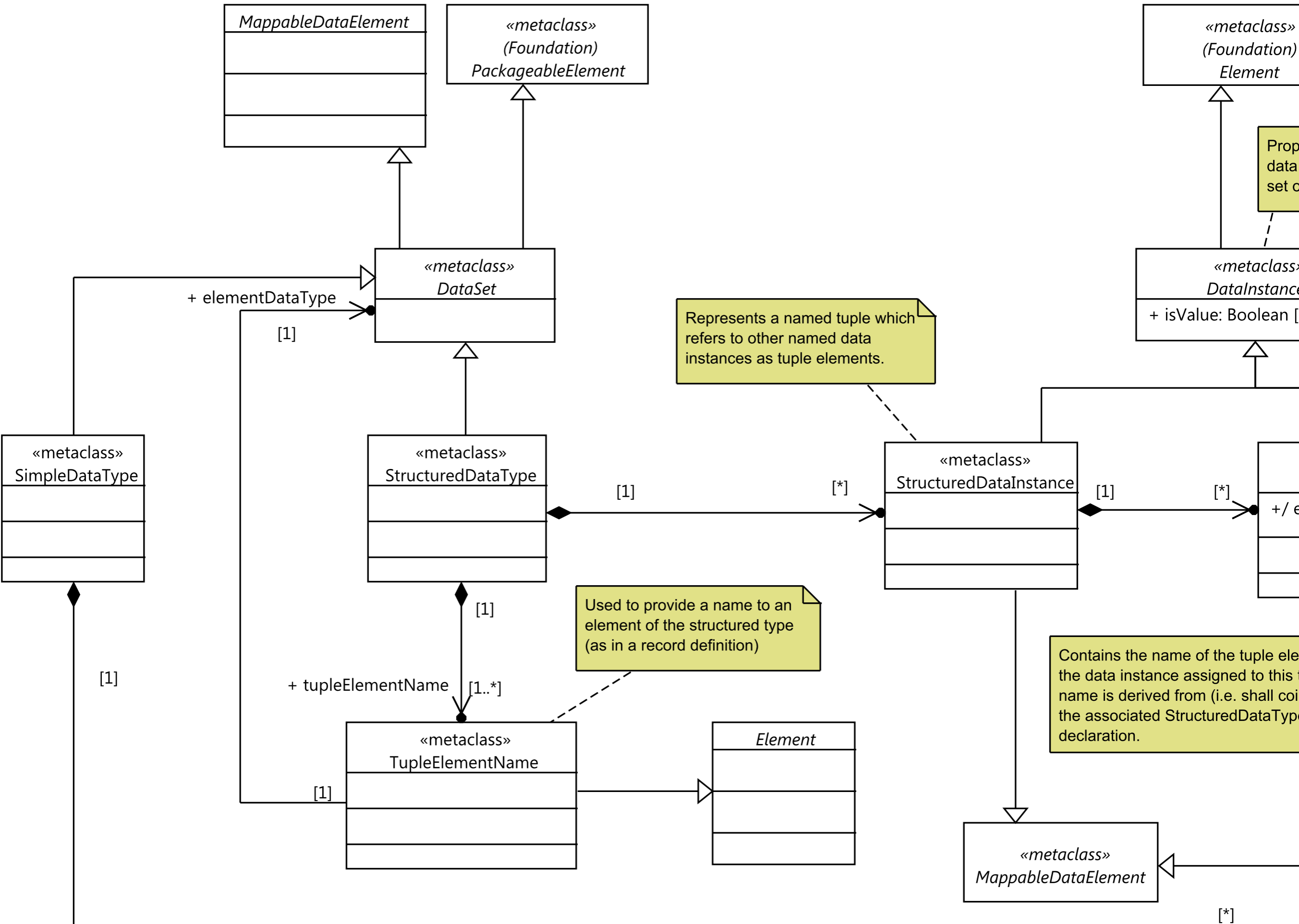
Task 1: Meta-Model

- Refinements of data and action concepts

  - variables and expressions

  - declaration and use of data related concepts

  - formalisation of actions (parameters, return types)

- Refinements of time related concepts

  - time observations and constraints

Task 1: Data Concepts

MappableDataElement

«metaclass»
(Foundation)
PackageableElement

«metaclass»
(Foundation)
Element

Prop
data
set o

«metaclass»
DataInstance

+ isValue: Boolean [

«metaclass»
DataSet

+ elementDataType

[1]

«metaclass»
SimpleDataType

«metaclass»
StructuredDataType

«metaclass»
StructuredDataInstance

[1]

[*]

+/ e

Represents a named tuple which
refers to other named data
instances as tuple elements.

[1]

[*]

Used to provide a name to an
element of the structured type
(as in a record definition)

+ tupleElementName

[1..*]

Contains the name of the tuple ele
the data instance assigned to this
name is derived from (i.e. shall coi
the associated StructuredDataTyp
declaration.

«metaclass»
TupleElementName

Element

[1]

«metaclass»
MappableDataElement

[1]

[*]

«metaclass»
(Foundation)
leElement

«metaclass»
(Foundation)
Element

Property "isValue" indicates whether the data instance reflects a single value or a set of possible values (template).

«metaclass»
DataInstance

+ isValue: Boolean [1]

Represents a named tuple which refers to other named data instances as tuple elements.
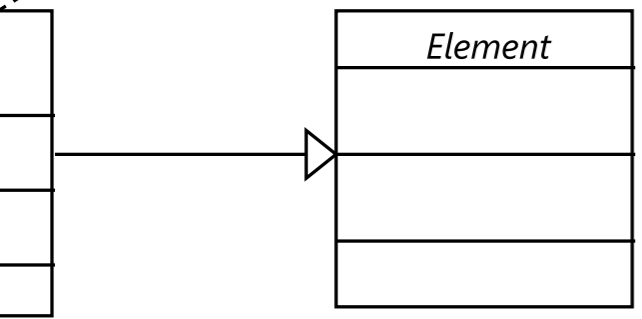
«metaclass»
StructuredDataInstance

[1]

[1]

«metaclass»
TupleElement

+/ elementName: String [0..1]

«metaclass»
AnyValue

[1]

[*]

[1]

[*]

Used to provide a name to an element of the structured type (as in a record definition)
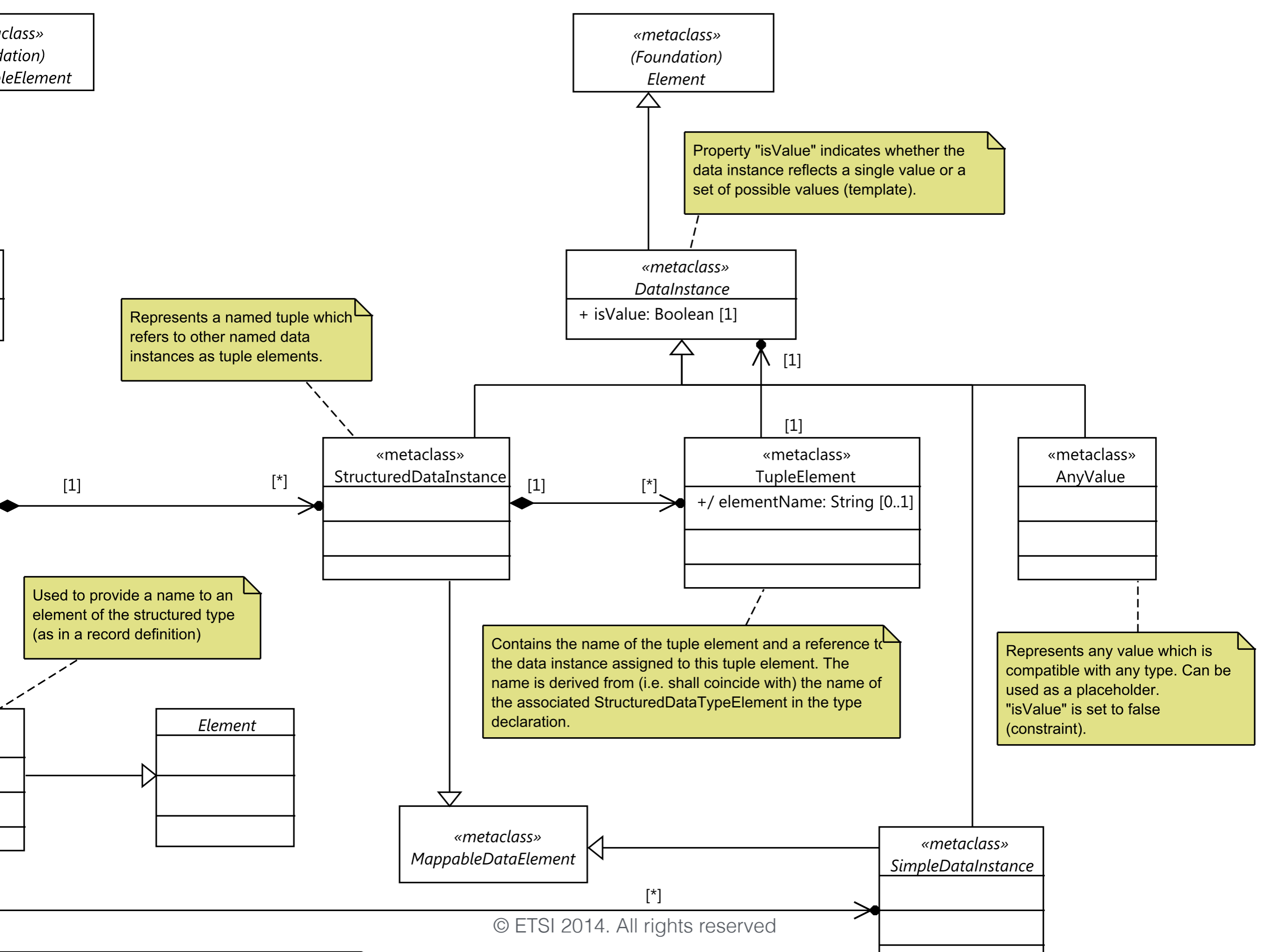
Contains the name of the tuple element and a reference to the data instance assigned to this tuple element. The name is derived from (i.e. shall coincide with) the name of the associated StructuredDataTypeElement in the type declaration.
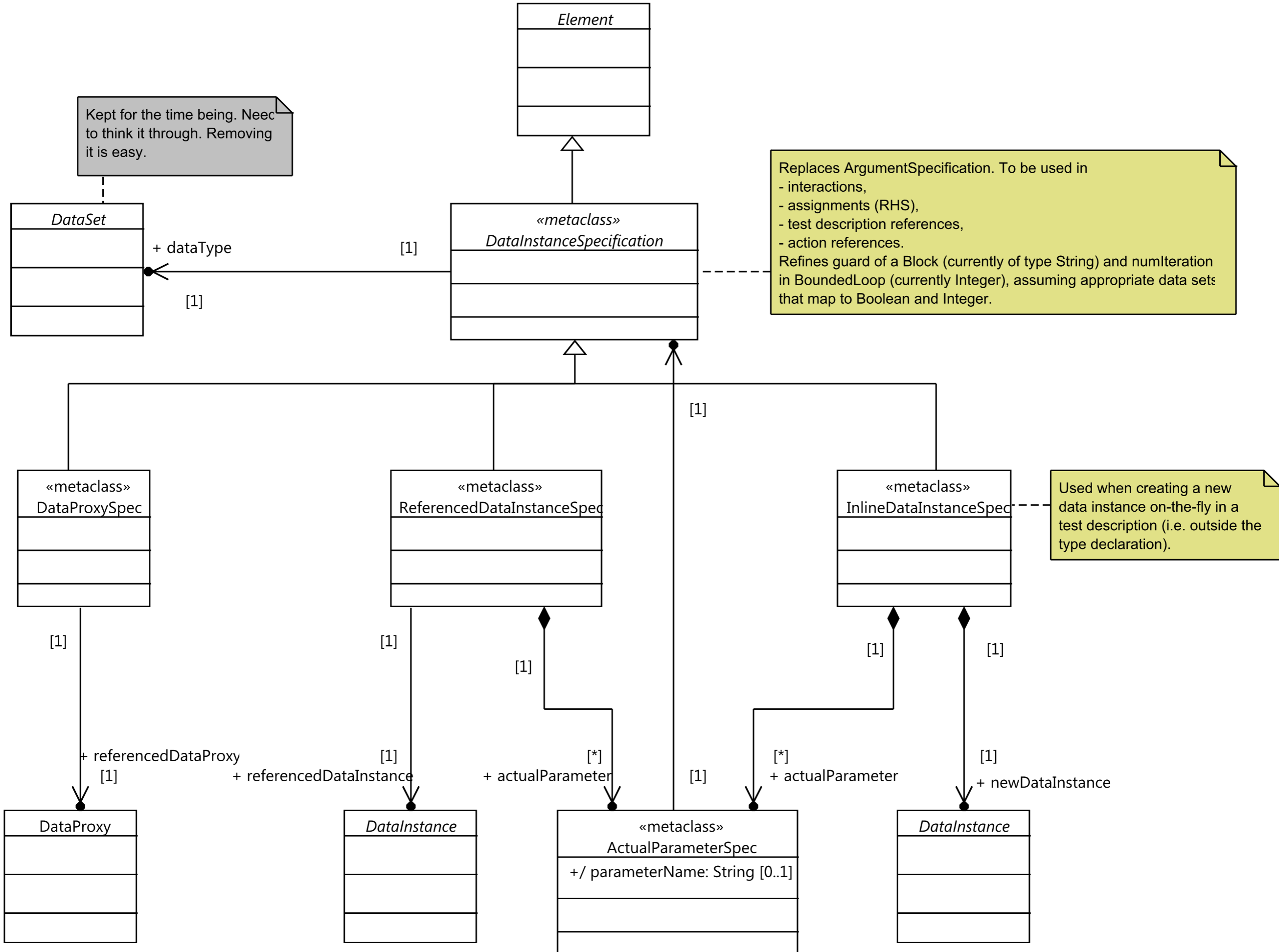
Represents any value which is compatible with any type. Can be used as a placeholder.
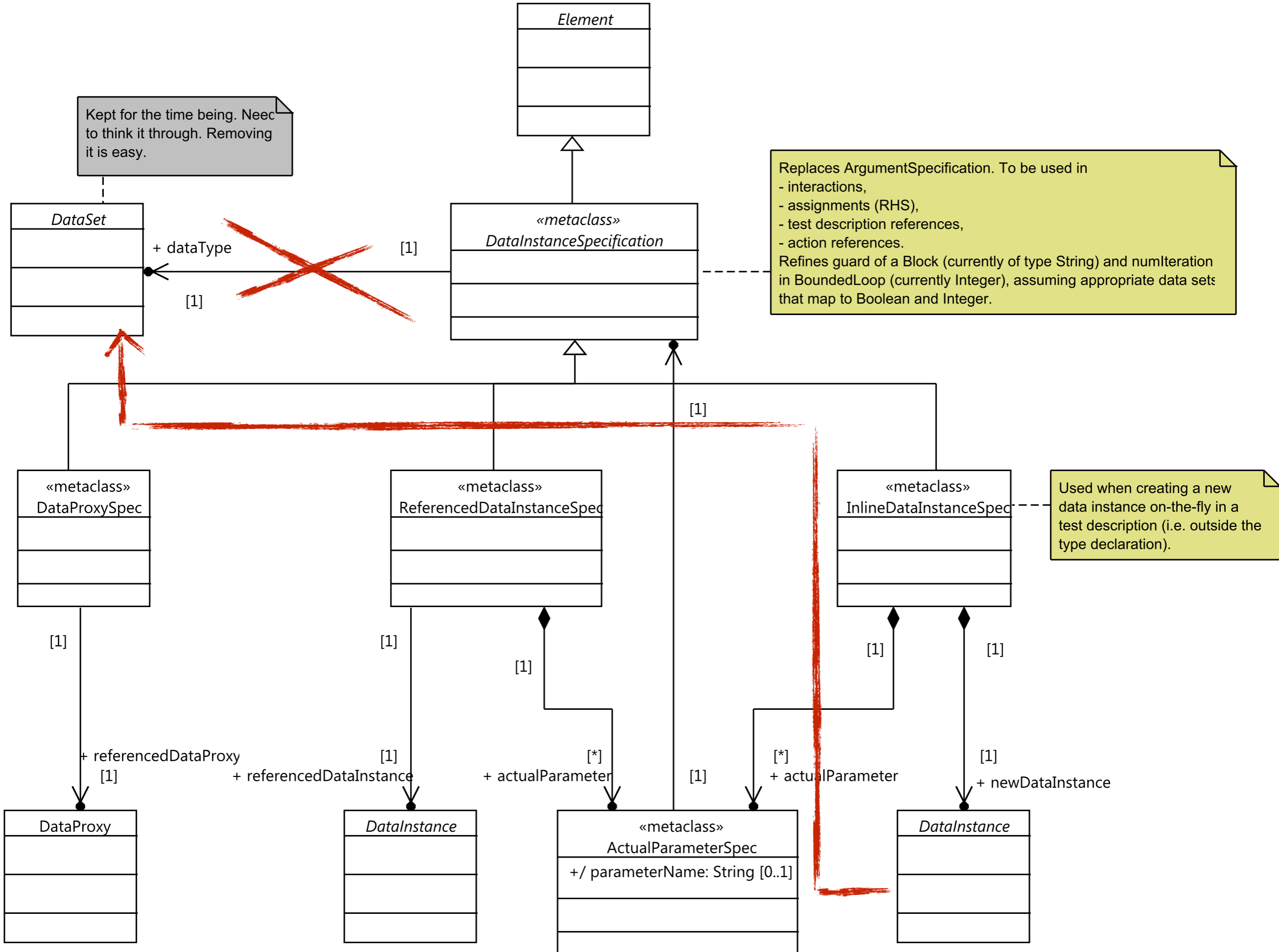"isValue" is set to false (constraint).

Element

«metaclass»
MappableDataElement

«metaclass»
SimpleDataInstance

[*]

Element

Kept for the time being. Need to think it through. Removing it is easy.

DataSet

+ dataType

[1]

[1]

«metaclass»
DataInstanceSpecification

Replaces ArgumentSpecification. To be used in
- interactions,
- assignments (RHS),
- test description references,
- action references.
Refines guard of a Block (currently of type String) and numIteration in BoundedLoop (currently Integer), assuming appropriate data sets that map to Boolean and Integer.

[1]

«metaclass»
DataProxySpec

«metaclass»
ReferencedDataInstanceSpec

«metaclass»
InlineDataInstanceSpec

Used when creating a new data instance on-the-fly in a test description (i.e. outside the type declaration).

[1]

[1]

[1]

[1]

[1]

[*]

[1]

[*]

[1]

+ referencedDataProxy
[1]

+ referencedDataInstance

+ actualParameter

+ actualParameter

+ newDataInstance

DataProxy

DataInstance

«metaclass»
ActualParameterSpec

+/ parameterName: String [0..1]

DataInstance

Element

Kept for the time being. Need to think it through. Removing it is easy.

DataSet

«metaclass»
DataInstanceSpecification

Replaces ArgumentSpecification. To be used in
- interactions,
- assignments (RHS),
- test description references,
- action references.
Refines guard of a Block (currently of type String) and numIteration in BoundedLoop (currently Integer), assuming appropriate data sets that map to Boolean and Integer.

+ dataType [1]

[1]

[1]

«metaclass»
DataProxySpec

«metaclass»
ReferencedDataInstanceSpec

«metaclass»
InlineDataInstanceSpec

Used when creating a new data instance on-the-fly in a test description (i.e. outside the type declaration).

[1]

[1]

[1]

[1]

[1]

[1]

[*]

[*]

[1]

+ referencedDataProxy
[1]

+ referencedDataInstance

+ actualParameter

+ actualParameter

+ newDataInstance

DataProxy

DataInstance

«metaclass»
ActualParameterSpec

+/ parameterName: String [0..1]

DataInstance

Represents component variables or formal parameters in functions, actions or test descriptions.
It can be bound to data instances, e.g. in assignments or when calling a function.

Moved from Behaviour to Data package, since it is really closer to data.

Because a function returns a value, it can be used whenever a data instance is allowed. Technically a function is not different from a structured data instance with parameters.

Task 1: Further Changes

- Test description: new root node for the behaviour

- Parallel: parallel blocks can have guards; removed constraint

- Action:

  - InlineAction as AtomicBehaviour with body:String

  - ActionMapping similar to data mapping

- Atomic behaviour: expressions and assignments

- Still a proposal! Extensive validation pending!

Task 2: Graphical Concrete Syntax

- Refinement and description of all elements that shall be represented graphically

  - data-related graphical elements to be proposed by 11.07.2014 due to delays in the meta-model updates

- Textual description of graphical elements

  - label aliases to be added for easier mapping

 - formalised description of labels within graphical elements

  - Informal mapping of graphical elements to meta-model elements

Task 2: Graphical Concrete Syntax

- Textual description of graphical elements

  - label aliases to be added for easier mapping

  - formalised description of labels within graphical elements

□

name:*type*   Gate Instance

«metaclass» (Test Architecture) GateInstance	+ gateInstance [*]	[1] + type

| «metaclass»
(Test Architecture)
GateType |
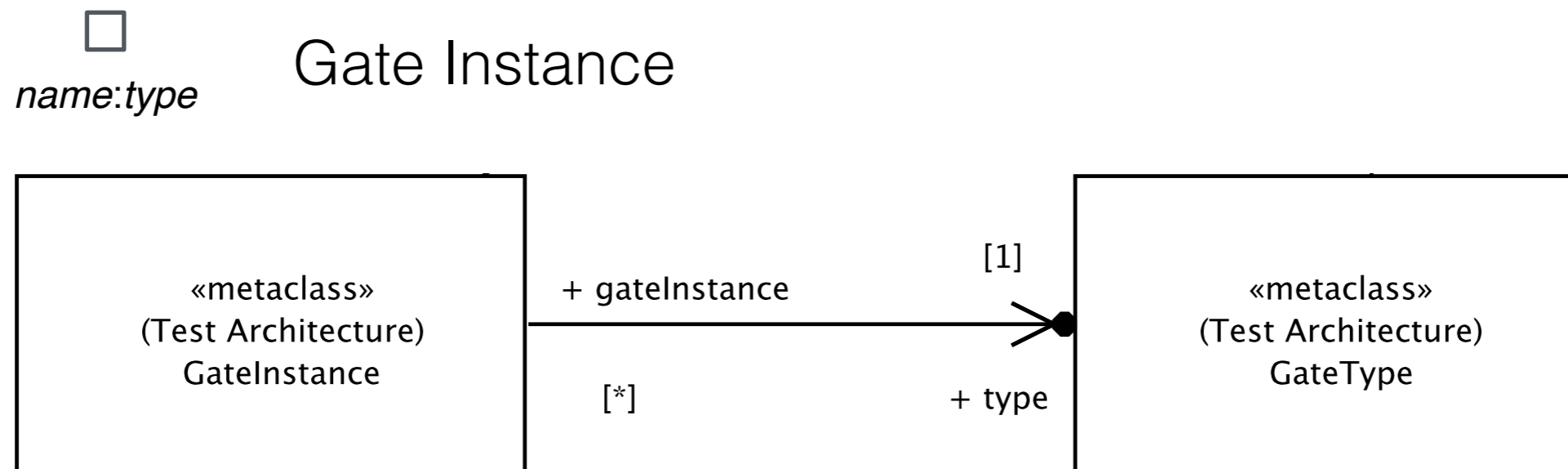
```
GateInstance.shape.name = GateInstance.model.name
GateInstance.shape.type = GateInstance.model.type.name
```

Task 2: Graphical Concrete Syntax

- New symbols

  - timer related symbols

- Diagrams

  - one type of diagram

  - multiple instances of the diagram can be created

  - diagrams contain diagram elements representing packageable elements

  - all other elements are represented within the containing element shapes

TimerOperations
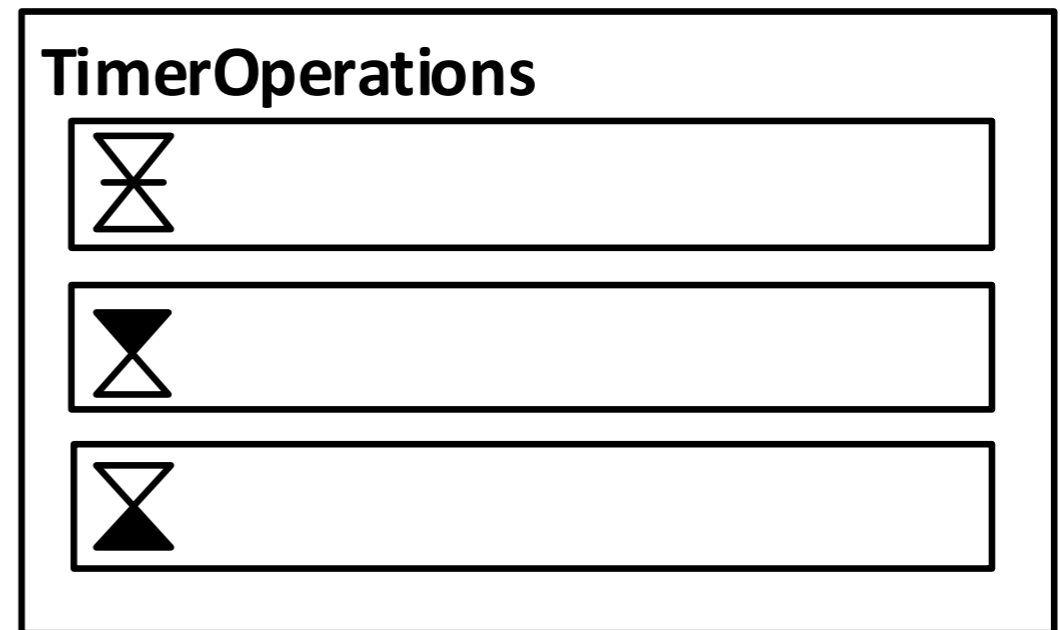
Task 2: Graphical Concrete Syntax

- Validation

  - (extended) IMS examples (from Part 1 and beyond)

  - examples from Siemens use cases

  - examples from Ericsson (later on)

Task 3: Exchange Format

- Introduction / scope / references

  - references and relationship to XMI, canonical XML, TDL

- Schema derivation

  - informal, based on references to relevant production rules from XMI

  - alternatively maintain complete set of production rules derived from XMI

  - limited expressiveness, additional information in Part 3 needed

  - additional refinements and restrictions where applicable

Task 3: Exchange Format

- Document Serialisation

  - document production rules

  - partial examples

- Annex

  - complete TDL schema

  - complete document examples (e.g. for examples in Part 1)

- Is EMF XMI schema sufficient? Is there is a need to go beyond?

Advanced Test Objective Specification

- New work item: Advanced Test Objective Specification with TDL

  - extension package for TDL, shall address ETSI's needs in particular

  - proposal accepted on 30.06.2014

    - minor delay due to technical (and possibly also coordination) issues

    - adjustment to start of work and deliverable schedules to be considered

  - funding for 20 days (total) + 5 days from CTI approved

    - budget even tighter than initially proposed (25 days)

    - needs to be taken into account for the scope of the deliverables

Further Activities

- Visibility

  - tutorial submitted and accepted for UCAAT 2014

  - website proposal prepared and submitted to STFLINK

Any Other Business?