



STIX^[TM] Version 1.2.1. Part 9: Course of Action

Committee Specification Draft 01 / Public Review Draft 01

06 November 2015

Specification URIs

This version:

<http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part9-coa/stix-v1.2.1-csprd01-part9-coa.docx>
(Authoritative)
<http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part9-coa/stix-v1.2.1-csprd01-part9-coa.html>
<http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part9-coa/stix-v1.2.1-csprd01-part9-coa.pdf>

Previous version:

N/A

Latest version:

<http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part9-coa.docx> (Authoritative)
<http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part9-coa.html>
<http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part9-coa.pdf>

Technical Committee:

OASIS Cyber Threat Intelligence (CTI) TC

Chair:

Richard Struse (Richard.Struse@HQ.DHS.GOV), DHS Office of Cybersecurity and Communications (CS&C)

Editors:

Sean Barnum (sbarnum@mitre.org), MITRE Corporation
Desiree Beck (dbeck@mitre.org), MITRE Corporation
Aharon Chernin (achernin@soltra.com), Soltra
Rich Piazza (rpiazza@mitre.org), MITRE Corporation

Additional artifacts:

This prose specification is one component of a Work Product that also includes:

- *STIX Version 1.2.1. Part 1: Overview.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part1-overview/stix-v1.2.1-csprd01-part1-overview.html>
- *STIX Version 1.2.1. Part 2: Common.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part2-common/stix-v1.2.1-csprd01-part2-common.html>
- *STIX Version 1.2.1. Part 3: Core.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part3-core/stix-v1.2.1-csprd01-part3-core.html>
- *STIX Version 1.2.1. Part 4: Indicator.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part4-indicator/stix-v1.2.1-csprd01-part4-indicator.html>
- *STIX Version 1.2.1. Part 5: TTP.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part5-ttp/stix-v1.2.1-csprd01-part5-ttp.html>
- *STIX Version 1.2.1. Part 6: Incident.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part6-incident/stix-v1.2.1-csprd01-part6-incident.html>

- *STIX Version 1.2.1. Part 7: Threat Actor.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part7-threat-actor/stix-v1.2.1-csprd01-part7-threat-actor.html>
- *STIX Version 1.2.1. Part 8: Campaign.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part8-campaign/stix-v1.2.1-csprd01-part8-campaign.html>
- *STIX Version 1.2.1. Part 9: Course of Action* (this document). <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part9-coa/stix-v1.2.1-csprd01-part9-coa.html>
- *STIX Version 1.2.1. Part 10: Exploit Target.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part10-exploit-target/stix-v1.2.1-csprd01-part10-exploit-target.html>
- *STIX Version 1.2.1. Part 11: Report.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part11-report/stix-v1.2.1-csprd01-part11-report.html>
- *STIX Version 1.2.1. Part 12: Default Extensions.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part12-extensions/stix-v1.2.1-csprd01-part12-extensions.html>
- *STIX Version 1.2.1. Part 13: Data Marking.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part13-data-marking/stix-v1.2.1-csprd01-part13-data-marking.html>
- *STIX Version 1.2.1. Part 14: Vocabularies.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part14-vocabularies/stix-v1.2.1-csprd01-part14-vocabularies.html>
- *STIX Version 1.2.1. Part 15: UML Model.* <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part15-uml-model/stix-v1.2.1-csprd01-part15-uml-model.html>
- UML Model Serialization: <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/uml-model/>

Related work:

This specification replaces or supersedes:

- *STIXTM 1.2 Course Of Action Specification (v1.2)* https://github.com/STIXProject/specifications/blob/version1.2/documents/pdf%20versions/STIX_COA_Draft.pdf

This specification is related to:

- *CybOXTM Version 2.1.1.* Work in progress. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cti-cybox
- *CybOXTM 2.1.* <https://cyboxproject.github.io/>

Abstract:

The Structured Threat Information Expression (STIX) framework defines nine core constructs and the relationships between them for the purposes of modeling cyber threat information and enabling cyber threat information analysis and sharing. This specification document defines the Course of Action construct, which conveys specific measures to be taken to address threats whether they are corrective or preventative to address Exploit Targets, or responsive to counter or mitigate the potential impacts of Incidents.

Status:

This document was last revised or approved by the OASIS Cyber Threat Intelligence (CTI) TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cti#technical.

TC members should send comments on this specification to the TC’s email list. Others should send comments to the TC’s public comment list, after subscribing to it by following the instructions at the “Send A Comment” button on the TC’s web page at <https://www.oasis-open.org/committees/cti/>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC’s web page (<https://www.oasis-open.org/committees/cti/ipr.php>).

Citation format:

When referencing this specification the following citation format should be used:

[STIX-v1.2.1-COA]

STIXTM Version 1.2.1. Part 9: Course of Action. Edited by Sean Barnum, Desiree Beck, Aharon Chernin, and Rich Piazza. 06 November 2015. OASIS Committee Specification Draft 01 / Public Review Draft 01. <http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part9-coa/stix-v1.2.1-csprd01-part9-coa.html>. Latest version: <http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part9-coa.html>.

Notices

Copyright © OASIS Open 2015. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full [Policy](#) may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

Portions copyright © United States Government 2012-2015. All Rights Reserved.

STIX[™], TAXII[™], AND CybOX[™] (STANDARD OR STANDARDS) AND THEIR COMPONENT PARTS ARE PROVIDED "AS IS" WITHOUT ANY WARRANTY OF ANY KIND, EITHER EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTY THAT THESE STANDARDS OR ANY OF THEIR COMPONENT PARTS WILL CONFORM TO SPECIFICATIONS, ANY

IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR FREEDOM FROM INFRINGEMENT, ANY WARRANTY THAT THE STANDARDS OR THEIR COMPONENT PARTS WILL BE ERROR FREE, OR ANY WARRANTY THAT THE DOCUMENTATION, IF PROVIDED, WILL CONFORM TO THE STANDARDS OR THEIR COMPONENT PARTS. IN NO EVENT SHALL THE UNITED STATES GOVERNMENT OR ITS CONTRACTORS OR SUBCONTRACTORS BE LIABLE FOR ANY DAMAGES, INCLUDING, BUT NOT LIMITED TO, DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES, ARISING OUT OF, RESULTING FROM, OR IN ANY WAY CONNECTED WITH THESE STANDARDS OR THEIR COMPONENT PARTS OR ANY PROVIDED DOCUMENTATION, WHETHER OR NOT BASED UPON WARRANTY, CONTRACT, TORT, OR OTHERWISE, WHETHER OR NOT INJURY WAS SUSTAINED BY PERSONS OR PROPERTY OR OTHERWISE, AND WHETHER OR NOT LOSS WAS SUSTAINED FROM, OR AROSE OUT OF THE RESULTS OF, OR USE OF, THE STANDARDS, THEIR COMPONENT PARTS, AND ANY PROVIDED DOCUMENTATION. THE UNITED STATES GOVERNMENT DISCLAIMS ALL WARRANTIES AND LIABILITIES REGARDING THE STANDARDS OR THEIR COMPONENT PARTS ATTRIBUTABLE TO ANY THIRD PARTY, IF PRESENT IN THE STANDARDS OR THEIR COMPONENT PARTS AND DISTRIBUTES IT OR THEM "AS IS."

Table of Contents

| | | |
|-------|---|----|
| 1 | Introduction | 7 |
| 1.1 | STIX ^[TM] Specification Documents | 7 |
| 1.2 | Document Conventions | 8 |
| 1.2.1 | Fonts..... | 8 |
| 1.2.2 | UML Package References | 8 |
| 1.2.3 | UML Diagrams..... | 9 |
| 1.2.4 | Property Table Notation | 10 |
| 1.2.5 | Property and Class Descriptions | 10 |
| 1.3 | Terminology | 11 |
| 1.4 | Normative References | 11 |
| 2 | Background Information | 12 |
| 2.1 | Course of Action-Related Component Data Models | 12 |
| 3 | STIX ^[TM] Course of Action Data Model..... | 14 |
| 3.1 | CourseOfActionVersionType Enumeration..... | 18 |
| 3.2 | StructuredCOAType Class | 18 |
| 3.3 | ObjectiveType Class..... | 18 |
| 3.4 | RelatedCOAsType Class..... | 20 |
| 4 | Conformance | 22 |
| | Appendix A. Acknowledgments | 23 |
| | Appendix B. Revision History..... | 25 |

1 Introduction

[All text is normative unless otherwise labeled]

The Structured Threat Information Expression (STIX^[TM]) framework defines nine top-level component data models: Observable¹, Indicator, Incident, TTP, ExploitTarget, CourseOfAction, Campaign, ThreatActor, and Report. This document serves as the specification for the STIX Course of Action data model.

As defined within the STIX language, a Course of Action (COA) characterizes a specific measure that could be taken in regard to a threat. These measures may be corrective or preventative to address Exploit Targets, or responsive to counter or mitigate the potential impacts of Incidents. They are typically cyber in nature but are not explicitly constrained to be so. More specifically, a Course of Action is fundamentally a characterization of the action through a title, description, type and structured observable parameters as well as contextual information such as objective, likely impact, likely cost, estimated efficacy and its relevant stage in cyber threat management (e.g., remedy of an ExploitTarget or response to an Incident).

In Section 1.1 we discuss additional specification documents, in Section 1.2 we provide document conventions, and in Section 1.3 we provide terminology. References are given in Section 1.4. In Section 2, we give background information necessary to fully understand the Course of Action data model. We present the Course of Action data model specification details in Section 3 and conformance information in Section 4.

1.1 STIX^[TM] Specification Documents

The STIX specification consists of a formal UML model and a set of textual specification documents that explain the UML model. Specification documents have been written for each of the key individual data models that compose the full STIX UML model.

The [STIX Version 1.2.1 Part 1: Overview](#) document provides a comprehensive overview of the full set of STIX data models, which in addition to the nine top-level component data models mentioned in the Introduction, includes a core data model, a common data model, a cross-cutting data marking data model, various extension data models, and a set of default controlled vocabularies. [STIX Version 1.2.1 Part 1: Overview](#) also summarizes the relationship of STIX to other languages, and outlines general STIX data model conventions.

Figure 1-1 illustrates the [set of specification documents](#) that are available. The color black is used to indicate the specification overview document, altered shading differentiates the overarching Core and Common data models from the supporting data models (vocabularies, data marking and default extensions), and the color white indicates the component data models. The solid grey color denotes the overall STIX Language UML model. This Course of Action specification document is highlighted in its associated color (see Section 1.2.3.3). For a list of all STIX documents and related information sources, please see [STIX Version 1.2.1 Part 1: Overview](#).



Figure 1-1. STIX[™] Language v1.2.1 specification documents

1.2 Document Conventions

The following conventions are used in this document.

1.2.1 Fonts

The following font and font style conventions are used in the document:

- Capitalization is used for STIX high level concepts, which are defined in [STIX Version 1.2.1 Part 1: Overview](#).

Examples: Indicator, Course of Action, Threat Actor

- The Courier New font is used for writing UML objects.

Examples: RelatedIndicatorsType, stixCommon:StatementType

Note that all high level concepts have a corresponding UML object. For example, the Course of Action high level concept is associated with a UML class named, CourseOfActionType.

- The *'italic'* font (with single quotes) is used for noting actual, explicit values for STIX Language properties. The *italic* font (without quotes) is used for noting example values.

Example: *'PackageIntentVocab-1.0'*, *high*, *medium*, *low*.

1.2.2 UML Package References

Each STIX data model is captured in a different UML package (e.g., Core package, Campaign package, etc.) where the packages together compose the full STIX UML model. To refer to a particular class of a specific package, we use the format `package_prefix:class`, where `package_prefix` corresponds to the appropriate UML package. [STIX Version 1.2.1 Part 1: Overview](#) contains a list of the packages used by the Course of Action data model, along with the associated prefix notations, descriptions, examples.

Note that in this specification document, we do not explicitly specify the package prefix for any classes that originate from the Course of Action data model.

1.2.3 UML Diagrams

This specification makes use of UML diagrams to visually depict relationships between STIX Language constructs. Note that the diagrams have been extracted directly from the full UML model for STIX; they have not been constructed purely for inclusion in the specification documents. Typically, diagrams are included for the primary class of a data model, and for any other class where the visualization of its relationships between other classes would be useful. This implies that there will be very few diagrams for classes whose only properties are either a data type or a class from the STIX Common data model. Other diagrams that are included correspond to classes that specialize a superclass and abstract or generalized classes that are extended by one or more subclasses.

In UML diagrams, classes are often presented with their attributes elided, to avoid clutter. The fully described class can usually be found in a related diagram. A class presented with an empty section at the bottom of the icon indicates that there are no attributes other than those that are visualized using associations.







1.2.3.1 Class Properties

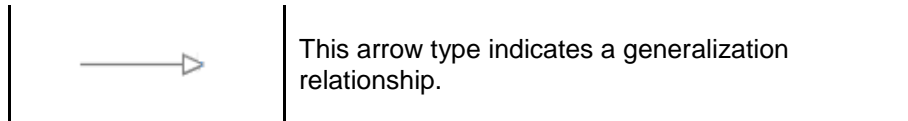
Generally, a class property can be shown in a UML diagram as either an attribute or an association (i.e., the distinction between attributes and associations is somewhat subjective). In order to make the size of UML diagrams in the specifications manageable, we have chosen to capture most properties as attributes and to capture only higher level properties as associations, especially in the main top-level component diagrams. In particular, we will always capture properties of UML data types as attributes. For example, properties of a class that are identifiers, titles, and timestamps will be represented as attributes.

1.2.3.2 Diagram Icons and Arrow Types

Diagram icons are used in a UML diagram to indicate whether a shape is a class, enumeration, or a data type, and decorative icons are used to indicate whether an element is an attribute of a class or an enumeration literal. In addition, two different arrow styles indicate either a directed association relationship (regular arrowhead) or a generalization relationship (triangle-shaped arrowhead). The icons and arrow styles we use are shown and described in [Table 1-1](#).

Table 1-1. UML diagram icons

| Icon | Description |
|---|--|
|  | This diagram icon indicates a class. If the name is in italics, it is an abstract class. |
|  | This diagram icon indicates an enumeration. |
|  | This diagram icon indicates a data type. |
|  | This decorator icon indicates an attribute of a class. The green circle means its visibility is public. If the circle is red or yellow, it means its visibility is private or protected. |
|  | This decorator icon indicates an enumeration literal. |
|  | This arrow type indicates a directed association relationship. |



1.2.3.3 Color Coding

The shapes of the UML diagrams are color coded to indicate the data model associated with a class. The colors used in the Course of Action specification are illustrated via exemplars in [Figure 1-2](#).



Figure 1-2. Data model color coding

1.2.4 Property Table Notation

Throughout Section 3, tables are used to describe the properties of each data model class. Each property table consists of a column of names to identify the property, a type column to reflect the datatype of the property, a multiplicity column to reflect the allowed number of occurrences of the property, and a description column that describes the property. Package prefixes are provided for classes outside of the Course of Action data model (see Section 1.2.2).

Note that if a class is a specialization of a superclass, only the properties that constitute the specialization are shown in the property table (i.e., properties of the superclass will not be shown). However, details of the superclass may be shown in the UML diagram.

1.2.5 Property and Class Descriptions

Each class and property defined in STIX is described using the format, “The X property verb Y.” For example, in the specification for the STIX Campaign, we write, “The id property specifies a globally unique identifier for the Campaign instance.” In fact, the verb “specifies” could have been replaced by any number of alternatives: “defines,” “describes,” “contains,” “references,” etc.

However, we thought that using a wide variety of verb phrases might confuse a reader of a specification document because the meaning of each verb could be interpreted slightly differently. On the other hand, we didn’t want to use a single, generic verb, such as “describes,” because although the different verb choices may or may not be meaningful from an implementation standpoint, a distinction could be useful to those interested in the modeling aspect of STIX.

Consequently, we have chosen to use the three verbs, defined as follows, in class and property descriptions:

| Verb | STIX Definition |
|-----------------|---|
| <u>captures</u> | Used to record and preserve information without implying anything about the structure of a class or property. Often used for properties that encompass general content. This is the least precise of the three verbs. |
| | <i>Examples:</i> The <u>source</u> property characterizes the source of the sighting information. Examples of details <u>captured</u> include identifying characteristics, time-related attributes, and a list of the tools used to collect the information. |

| | |
|----------------------|---|
| | The <code>Description</code> property <u>captures</u> a textual description of the Indicator. |
| <u>characterizes</u> | Describes the distinctive nature or features of a class or property. Often used to describe classes and properties that themselves comprise one or more other properties. |
| | <p><i>Examples:</i></p> <p>The <code>Confidence</code> property <u>characterizes</u> the level of confidence in the accuracy of the overall content captured in the Incident.</p> <p>The <code>ActivityType</code> class <u>characterizes</u> basic information about an activity a defender might use in response to a Campaign.</p> |
| <u>specifies</u> | Used to clearly and precisely identify particular instances or values associated with a property. Often used for properties that are defined by a controlled vocabulary or enumeration; typically used for properties that take on only a single value. |
| | <p><i>Example:</i></p> <p>The <code>version</code> property <u>specifies</u> the version identifier of the STIX Campaign data model used to capture the information associated with the Campaign.</p> |

1.3 Terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

1.4 Normative References

- [RFC2119] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, BCP 14, RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>.

2 Background Information

In this section, we provide high level information about the Course of Action data model that is necessary to fully understand the Course of Action data model specification details given in Section 3.

2.1 Course of Action-Related Component Data Models

As will be explicitly detailed in Section 3, a STIX Course of Action leverages the Observables data model (as indicated by the outward-oriented arrow) which is defined with the CybOX Language. **Figure 2-1** illustrates the relationship between the Course of Action and the other core constructs. As stated in Section 1.1, each of these components is defined in a separate specification document.

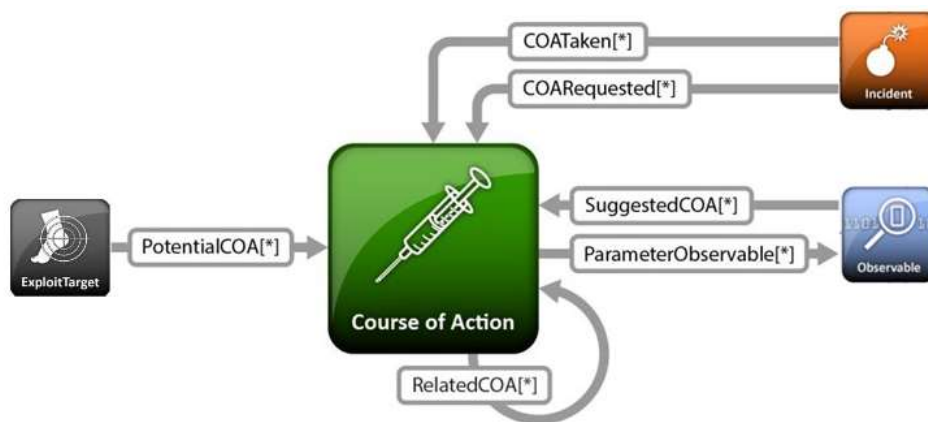


Figure 2-1. High level view of the Course of Action data model

In this section, we give a high level summary of the relationship between the Course of Action data model and the other components to which a Course of Action may refer. We also make note of the fact that the Course of Action data model can be self-referential. Other relationships are defined in the specification of the component that they originate from.

- **Course of Action**

The Course of Action data model is self-referential, enabling one Course of Action to reference other Course of Actions that are asserted to be related. Self-referential relationships between Courses of Action may indicate general associativity or can be used to indicate relationships between different versions of the same Course of Action.

- **Observable**

A STIX Observable (as defined with the [CybOX Language](#)) represents stateful properties or measurable events pertinent to the operation of computers and networks. Implicit in this is a practical need for descriptive capability of two forms of observables: “observable instances” and “observable patterns.” Observable instances represent actual specific observations that took place in the cyber domain. The property details of this observation are specific and unambiguous. Observable patterns represent conditions for a potential observation that may occur in the future or may have already occurred and exists in a body of observable instances. These conditions may be anything from very specific concrete patterns that would match very specific observable instances to more abstract generalized patterns that have the potential to match against a broad range of potential observable instances.

The Course of Action data model leverages the Observable data model to specify observable patterns to be used as structured parameters for the action specified in the `Type` property (e.g. a structured characterization of an outbound network connection to a particular IP address that when combined with a `Type="Block"` unambiguously describes an action of blocking such traffic).

3 STIXTM Course of Action Data Model

The primary class of the STIX Course of Action package is the `CourseOfActionType` class, which characterizes a cyber threat-relevant course of action through informative (title and description), formally structured (type and parameter observables) and contextual (objective, efficacy, impact, cost) properties. Similar to the primary classes of all of the component data models in STIX, the `CourseOfActionType` class extends a base class defined in the STIX Common data model; more specifically, it extends the `CourseOfActionBaseType` base class, which provides the essential identifier (`id`) and identifier reference (`idref`) properties.

The relationship between the `CourseOfActionType` class and the `CourseOfActionBaseType` base class, as well as the properties of the `CourseOfActionType` class, are illustrated in the UML diagram given in [Figure 3-1](#).

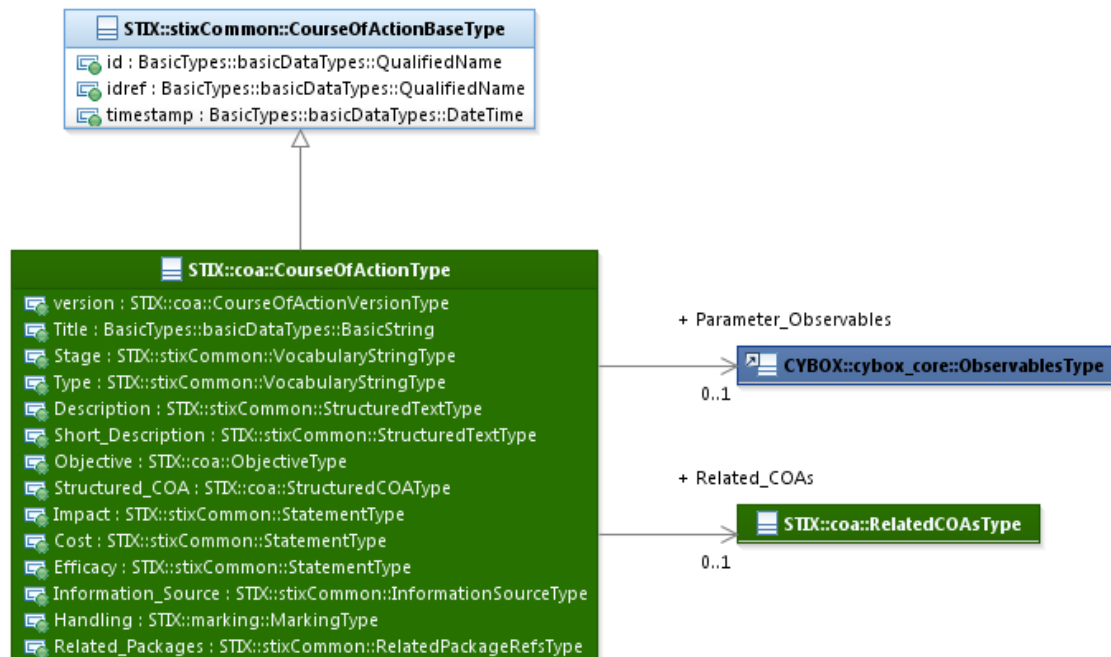


Figure 3-1. UML diagram of the `CourseOfActionType` class

The property table, which includes property descriptions and corresponds to the UML diagram given in [Figure 3-1](#), is provided in [Table 3-1](#).

All classes defined in the Course of Action data model are described in detail in Section [3.1](#) through Section [3.4](#). Details are not provided for classes defined in non-Course of Action data models; instead, the reader is referred to the corresponding data model specification as indicated by the package prefix specified in the Type column of the table.

Table 3-1. Properties of the *CourseOfActionType* class

| Name | Type | Multiplicity | Description |
|----------------|-------------------------------------|--------------|--|
| version | CourseOfActionVersionType | 0..1 | The <code>version</code> property specifies the version number of the STIX Course of Action data model for STIX v1.2.1 used to capture the information associated with the Course of Action. |
| Title | basicDataTypes:BasicString | 0..1 | The <code>Title</code> property captures a title for the Course of Action and reflects what the content producer thinks the Course of Action as a whole should be called. The <code>Title</code> property is typically used by humans to reference a particular Course of Action; however, it is not suggested for correlation. |
| Stage | stixCommon: VocabularyStringType | 0..1 | The <code>Stage</code> property specifies what stage in the cyber threat management lifecycle this Course of Action is relevant to. Examples of potential stages include <i>remedy</i> and <i>response</i> (these specific values are only provided to help explain the property: they are neither recommended values nor necessarily part of any existing vocabulary). The content creator may choose any arbitrary value or may constrain the set of possible values by referencing an externally-defined vocabulary or leveraging a formally defined vocabulary extending from the <code>stixCommon:ControlledVocabularyStringType</code> class. The STIX default vocabulary class for use in the property is ' <i>COAStageVocab-1.0</i> '. |
| Type | stixCommon: VocabularyStringType | 0..1 | The <code>Type</code> property specifies the type of action to be taken. Examples of potential types include <i>redirection</i> , <i>eradication</i> and <i>public disclosure</i> (these specific values are only provided to help explain the property: they are neither recommended values nor |

| | | | |
|------------------------------|--|------|--|
| | | | necessarily part of any existing vocabulary). The content creator may choose any arbitrary value or may constrain the set of possible values by referencing an externally-defined vocabulary or leveraging a formally defined vocabulary extending from the <code>stixCommon:ControlledVocabularyStringType</code> class. The STIX default vocabulary class for use in the property is ' <code>CourseOfActionTypeVocab-1.0</code> '. |
| Description | <code>stixCommon:StructuredTextType</code> | 0..* | The <code>Description</code> property captures a textual description of the Course of Action. Any length is permitted. Optional formatting is supported via the <code>structuring_format</code> property of the <code>StructuredTextType</code> class. |
| Short_Description | <code>stixCommon:StructuredTextType</code> | 0..* | The <code>Short_Description</code> property captures a short textual description of the Course of Action. This property is secondary and should only be used if the <code>Description</code> property is already populated and another, shorter description is available. |
| Objective | <code>ObjectiveType</code> | 0..1 | The <code>Objective</code> property characterizes the results that this Course of Action is intended to achieve. |
| Parameter_Observables | <code>cybox:ObservablesType</code> | 0..1 | The <code>Parameter_Observables</code> property enables the specification of structured technical parameters to this Course of Action expressed using the CybOX Language. It is intended that the combination of the Course of Action Type and the <code>Parameter_Observables</code> could be used to define unambiguous and potentially automated courses of action. |
| Structured_COA | <code>StructuredCOAType</code> | 0..1 | The <code>Structured_COA</code> property characterizes an alternative actionable structured representation for the Course of Action potentially for automated consumption and implementation. Its underlying abstract class MUST be extended to enable the expression of a structured Course of Action. |
| Impact | <code>stixCommon:StatementType</code> | 0..1 | The <code>Impact</code> property characterizes the estimated impact of applying a Course of Action to achieve its targeted objective, which includes a <code>Value</code> property that specifies the level of impact. Examples of potential levels include <i>high</i> , <i>medium</i> , and <i>low</i> (these specific values |

| | | | |
|---------------------------|---|------|---|
| | | | are only provided to help explain the <code>Value</code> property: they are neither recommended values nor necessarily part of any existing vocabulary). The content creator may choose any arbitrary value or may constrain the set of possible levels by referencing an externally-defined vocabulary. The STIX default vocabulary class for use in the <code>Value</code> property is <i>'HighMediumLowVocab-1.0.'</i> |
| Cost | <code>stixCommon:StatementType</code> | 0..1 | The <code>Cost</code> property characterizes the estimated cost for applying a <code>Course of Action</code> to achieve its targeted objective, which includes a <code>Value</code> property that specifies the level of cost. Examples of potential levels include <i>high</i> , <i>medium</i> , and <i>low</i> (these specific values are only provided to help explain the <code>Value</code> property: they are neither recommended values nor necessarily part of any existing vocabulary). The content creator may choose any arbitrary value or may constrain the set of possible levels by referencing an externally-defined vocabulary. The STIX default vocabulary class for use in the <code>Value</code> property is <i>'HighMediumLowVocab-1.0.'</i> |
| Efficacy | <code>stixCommon:StatementType</code> | 0..1 | The <code>Efficacy</code> property characterizes a measure of the likely effectiveness of a <code>Course of Action</code> to achieve its targeted objective, which includes a <code>Value</code> property that specifies the level of effectiveness. Examples of potential levels include <i>high</i> , <i>medium</i> , and <i>low</i> (these specific values are only provided to help explain the <code>Value</code> property: they are neither recommended values nor necessarily part of any existing vocabulary). The content creator may choose any arbitrary value or may constrain the set of possible levels by referencing an externally-defined vocabulary. The STIX default vocabulary class for use in the <code>Value</code> property is <i>'HighMediumLowVocab-1.0.'</i> |
| Information_Source | <code>stixCommon:InformationSourceType</code> | 0..1 | The <code>Information_Source</code> property characterizes the source of the <code>Course of Action</code> information. Examples of details captured include identifying characteristics, time-related attributes, and a list of tools used to collect the information. |
| Handling | <code>marking:MarkingType</code> | 0..1 | The <code>Handling</code> property specifies the appropriate data handling markings for the properties of this <code>Course of Action</code> . The marking scope is limited to the <code>Course of Action</code> and the content it contains. |

| | | | |
|-------------------------|--|------|---|
| | | | Note that data handling markings can also be specified at a higher level. |
| Related_COAs | RelatedCOAsType | 0..1 | The <code>Related_COAs</code> property specifies a set of one or more other Course of Actions related to this Course of Action. |
| Related_Packages | stixCommon: RelatedPackagesRefsType | 0..1 | The <code>Related_Packages</code> property specifies a set of one or more STIX Packages that are related to the Course of Action. DEPRECATED: This property is deprecated and will be removed in the next major version of STIX. Its use is strongly discouraged except for legacy applications. |

3.1 CourseOfActionVersionType Enumeration

The `CourseOfActionVersionType` enumeration is an inventory of all versions of the Course of Action data model for STIX Version 1.2.1. The enumeration literals are given in [Table 3-2](#).

Table 3-2. Literals of the `CourseOfActionVersionType` enumeration

| Enumeration Literal | Description |
|---------------------|---|
| stix-1.2.1 | Course of Action data model for STIX v1.2.1 |

3.2 StructuredCOAType Class

The `StructuredCOAType` class enables the specification of an alternative actionable structured representation for the Course of Action potentially for automated consumption and implementation. The `StructuredCOAType` class is an abstract class and is intended to be extended via a subclass to enable the expression of any structured course of actions. STIX has provided support for passing proprietary or externally defined structured courses of action using the `GenericStructuredCOAType` class (see [STIX Version 1.2.1 Part 10: Exploit Target](#)).

3.3 ObjectiveType Class

The `ObjectiveType` class characterizes the results that this Course of Action is intended to achieve.

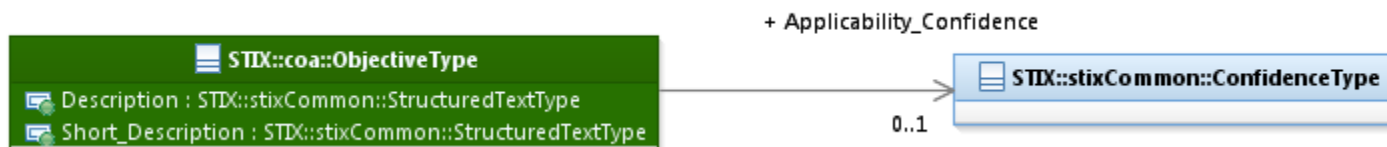


Figure 3-2. UML diagram of the *ObjectiveType* class

The property table, which includes property descriptions and corresponds to the UML diagram given in [Figure 3-2](#), is provided in [Table 3-3](#).

Table 3-3. Properties of the *ObjectiveType* class

| Name | Type | Multiplicity | Description |
|---------------------------------|-------------------------------|--------------|---|
| Description | stixCommon:StructuredTextType | 0..1 | The <code>Description</code> property captures a textual description of the objective of this Course of Action. Any length is permitted. Optional formatting is supported via the <code>structuring_format</code> property of the <code>StructuredTextType</code> class. |
| Short_Description | stixCommon:StructuredTextType | 0..1 | The <code>Short_Description</code> property captures a short textual description of the objective of this Course of Action. This property is secondary and should only be used if the <code>Description</code> property is already populated and another, shorter description is available. |
| Applicability_Confidence | stixCommon:ConfidenceType | 0..1 | The <code>Applicability_Confidence</code> property characterizes the level of confidence in the asserted applicability of the suggested Course of Action for its targeted objective. |

3.4 RelatedCOAsType Class

The `RelatedCOAsType` class specifies a set of one or more other Course of Actions asserted to be related to this Course of Action and therefore is a self-referential relationship. It extends the `GenericRelationshipListType` superclass defined in the STIX Common data model, which specifies the scope (whether the elements of the set are related individually or as a group).

The UML diagram corresponding to the `RelatedCOAsType` class is shown in [Figure 3-3](#).

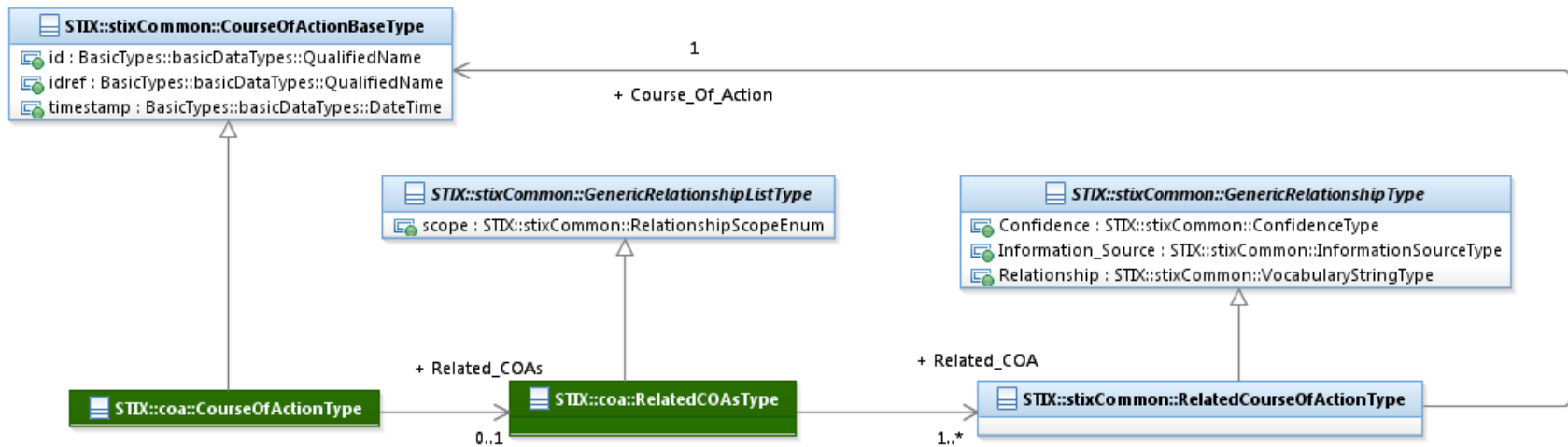


Figure 3-3. UML diagram of the `RelatedCOAsType` class

The property table given in [Table 3-4](#) corresponds to the UML diagram shown in [Figure 3-3](#).

Table 3-4. Properties of the `RelatedCOAsType` class

| Name | Type | Multiplicity | Description |
|---------------------------------|--|--------------|--|
| Related_Course of Action | stixCommon: RelatedCourseOfActionType | 1..* | The <code>Related_COA</code> property specifies another Course of Action associated with this Course of Action and characterizes the relationship between the Courses of Action by capturing information such as the level of confidence that the Courses of |

| | | | |
|--|--|--|---|
| | | | Actions are related, the source of the relationship information, and type of the relationship. A relationship between Courses of Action may represent assertions of general associativity or different versions of the same Course of Action. |
|--|--|--|---|

4 Conformance

Implementations have discretion over which parts (components, properties, extensions, controlled vocabularies, etc.) of STIX they implement (e.g., Indicator/Suggested_COAs).

[1] Conformant implementations must conform to all normative structural specifications of the UML model or additional normative statements within this document that apply to the portions of STIX they implement (e.g., Implementers of the entire TTP component must conform to all normative structural specifications of the UML model or additional normative statements within this document regarding the TTP component).

[2] Conformant implementations are free to ignore normative structural specifications of the UML model or additional normative statements within this document that do not apply to the portions of STIX they implement (e.g., Non-implementers of any particular properties of the TTP component are free to ignore all normative structural specifications of the UML model or additional normative statements within this document regarding those properties of the TTP component).

The conformance section of this document is intentionally broad and attempts to reiterate what already exists in this document. The STIX 1.2 Specifications, which this specification is based on, did not have a conformance section. Instead, the STIX 1.2 Specifications relied on normative statements and the non-mandatory implementation of STIX profiles. STIX 1.2.1 represents a minimal change from STIX 1.2, and in that spirit no requirements have been added, modified, or removed by this section.

Appendix A. Acknowledgments

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Dean Thompson, Australia and New Zealand Banking Group (ANZ Bank)
Bret Jordan, Blue Coat Systems, Inc.
Adnan Baykal, Center for Internet Security (CIS)
Jyoti Verma, Cisco Systems
Liron Schiff, Comilion (mobile) Ltd.
Jane Ginn, Cyber Threat Intelligence Network, Inc. (CTIN)
Richard Struse, DHS Office of Cybersecurity and Communications (CS&C)
Marlon Taylor, DHS Office of Cybersecurity and Communications (CS&C)
David Eilken, Financial Services Information Sharing and Analysis Center (FS-ISAC)
Sarah Brown, Fox-IT
Ryusuke Masuoka, Fujitsu Limited
Eric Burger, Georgetown University
Jason Keirstead, IBM
Paul Martini, iboss, Inc.
Jerome Athias, Individual
Terry MacDonald, Individual
Alex Pinto, Individual
Patrick Maroney, Integrated Networking Technologies, Inc.
Wouter Bolsterlee, Intelworks BV
Joep Gommers, Intelworks BV
Sergey Polzunov, Intelworks BV
Rutger Prins, Intelworks BV
Andrei Sirghi, Intelworks BV
Raymon van der Velde, Intelworks BV
Jonathan Baker, MITRE Corporation
Sean Barnum, MITRE Corporation
Desiree Beck, MITRE Corporation
Mark Davidson, MITRE Corporation
Ivan Kirillov, MITRE Corporation
Jon Salwen, MITRE Corporation
John Wunder, MITRE Corporation
Mike Boyle, National Security Agency
Jessica Fitzgerald-McKay, National Security Agency
Takahiro Kakumaru, NEC Corporation
John-Mark Gurney, New Context Services, Inc.
Christian Hunt, New Context Services, Inc.
Daniel Riedel, New Context Services, Inc.
Andrew Storms, New Context Services, Inc.
John Tolbert, Queralt, Inc.
Igor Baikalov, Securonix
Bernd Grobauer, Siemens AG
Jonathan Bush, Soltra
Aharon Chernin, Soltra
Trey Darley, Soltra
Paul Dion, Soltra
Ali Khan, Soltra
Natalie Suarez, Soltra
Cedric LeRoux, Splunk Inc.
Brian Luger, Splunk Inc.
Crystal Hayes, The Boeing Company
Brad Butts, U.S. Bank

Mona Magathan, U.S. Bank
Adam Cooper, United Kingdom Cabinet Office
Mike McLellan, United Kingdom Cabinet Office
Chris O'Brien, United Kingdom Cabinet Office
Julian White, United Kingdom Cabinet Office
Anthony Rutkowski, Yaana Technologies, LLC

The authors would also like to thank the larger STIX Community for its input and help in reviewing this document.

Appendix B. Revision History

| Revision | Date | Editor | Changes Made |
|----------|----------------|--|------------------------------------|
| wd01 | 21 August 2015 | Sean Barnum Desiree Beck Aharon Chernin Rich Piazza | Initial transfer to OASIS template |

Notes _____

¹ The CybOX Observable data model is actually defined in the [CybOX Language](#), not in STIX.