STIX[TM] Version 1.2.1. Part 15: UML Model

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/part15-uml-model/stix-v1.2.1-csprd01-part15-uml-model.docx (Authoritative)
http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part15-uml-model/stix-v1.2.1-csprd01-part15-uml-model.html
http://docs.oasis-open.org/cti/stix/v1.2.1/csprd01/part15-uml-model/stix-v1.2.1-csprd01-part15-uml-model.pdf

Previous version:

N/A

Latest version:

http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part15-uml-model.docx (Authoritative)
http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part15-uml-model.html
http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part15-uml-model.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*. 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* (this document). 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 is related to:

- *CybOX[TM] Version 2.1.1.* Work in progress. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cti-cybox
- *CybOX[TM] 2.1.* https://cyboxproject.github.io/

Abstract:

The Structured Threat Information Expression (STIX) is a collaborative, community-driven effort to define and develop a framework for expressing cyber threat information to enable cyber threat information sharing and cyber threat analysis. The STIX framework comprises a collection of extensible component specifications along with an overarching core specification and supporting specifications. This document describes the use of UML to create a data model for STIX.

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-UML-Model]

STIX[TM] Version 1.2.1. Part 15: UML Model. 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/part15-uml-model/stix-v1.2.1-csprd01-part15-uml-model.html. Latest version: http://docs.oasis-open.org/cti/stix/v1.2.1/stix-v1.2.1-part15-uml-model.html.

Notices

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.

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

Table of Contents

1 Introduction 7
1.1 STIX[TM] Specification Documents 7
1.2 Document Conventions 8
1.2.1 Fonts 8
1.3 Terminology 8
1.4 Normative References 8
1.5 Non-Normative References 8
2 UML Model Artifact 9
3 Data Model Conventions 10
3.1 UML Packages 10
3.2 Naming Conventions 13
3.3 UML Diagrams 13
3.3.1 Class Properties 14
3.3.2 Diagram Icons and Arrow Types 14
3.3.3 Color Coding 15
4 Conformance 16
Appendix A. Acknowledgments 17
Appendix B. Revision History 19

1 Introduction

[All text is normative unless otherwise labeled]

The objective of the Structured Threat Information Expression (STIX[TM]) effort is to specify the structure and semantics of a language for capturing and characterizing cyber threat information. The normative specification of the language structure is defined in the form of a formal UML model and a set of textual specification documents that explain the UML model. The set of textual specification documents also provides clarification of language semantics that the UML model is unable to convey.

This specification document provides brief summary information on the form and use of the STIX Language UML model. In addition to this textual specification document, *STIX Version 1.2.1 Part 15: UML Model* consists of an actual digital serialization of the UML model and a set of relevant UML diagrams extracted from the UML model and used throughout the STIX Language specification.

In Section **1.1** we discuss the additional specification documents, in Section **1.2** we provide document conventions, and in Section **1.3** we provide terminology. References are given in Sections **1.4** and **1.5**. In Section **2**, we give summary information on the form of the digitally serialized UML model artifact, and in Section **3** we provide general information and conventions for how the UML model is used to define the individual data models. Conformance information is provided in Section **4**.

1.1 STIX[TM] Specification Documents

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 data models models (Observable[1], Indicator, Incident, TTP, ExploitTarget, CourseOfAction, Campaign, ThreatActor, and Report), 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. This STIX Language UML Model specification document is shown in yellow. For a list of all STIX documents and related information sources, please see *STIX Version 1.2.1 Part 1: Overview*.

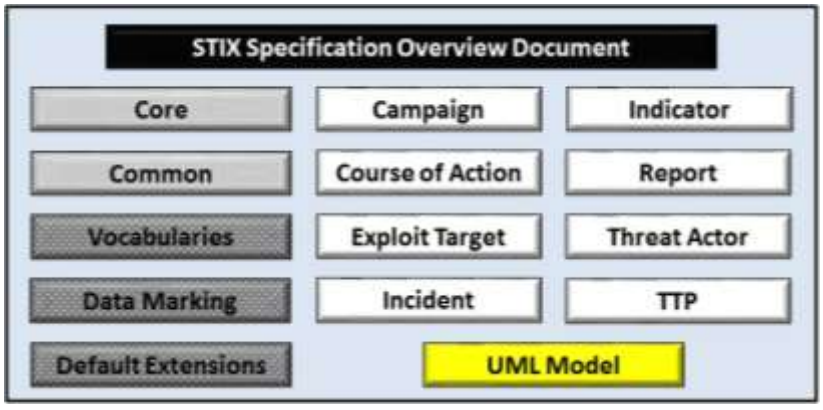

Figure 1-1. STIX[TM] Language v1.2.1 specification documents

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.

  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*

1.3 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.

1.5 Non-Normative References

[GitHub-IO] STIX – Structured Threat Information Expression | STIX Project Documentation. (n.d.). The MITRE Corporation. [Online]. Available: http://stixproject.github.io/. Accessed Aug. 23, 2015.

[STIX-W] Barnum, S., "Standardizing Cyber Threat Intelligence with the Structured Threat Information eXpression (STIX™)," The MITRE Corporation, Bedford MA, Feb. 20, 2014. [Online]. Available: http://stixproject.github.io/getting-started/whitepaper/.

[UML-2.4.1] Documents associated with Unified Modeling Language (UML), V2.4.1. (Aug. 2011). The Object Management Group (OMG). [Online]. Available: http://www.omg.org/spec/UML/2.4.1/.

[XMI] Documents associated with XMI Version 2.1. (September 2005). The Object Management Group (OMG). [Online]. Available: http://www.omg.org/spec/XMI/2.1/

[PNG] Portable Network Graphics (PNG) Specification (November 2003). The World Wide Web Consortium (W3C). [Online]. Available: http://www.w3.org/TR/PNG/

2 UML Model Artifact

The STIX UML model is formally represented in the form of a digital serialization using the XML Metadata Interchange (XMI) language. The XMI language is intended to be an open standardized form supporting the expression of UML models in a non-proprietary manner. In reality, many UML modeling tools tend to include some proprietary elements in their XMI output. The STIX UML model was produced using Rational Software Architect (RSA) version 9.1, a product of the IBM Corporation. Effort has been made to minimize the level of proprietary content (from the RSA tool) in the XMI serialization but it should be noted that some portion may still remain.

For the broadest possible interoperability between UML tools the model is provided as an XMI serialization using UML2.2/XMI2.1 [XMI] containing only the model and not the diagrams. A set of relevant UML diagrams, extracted from the UML model and leveraged throughout the STIX Language specification documents, is also provided in a rastered (portable network graphics [PNG]) form.

In addition, for those with tools that can import the more complete RSA tool native .EMX format the model with embedded diagrams is also provided in this form.

3 Data Model Conventions

The following general information and conventions are used to define the individual data models in UML.

3.1 UML Packages

Each STIX data model is captured in a different UML package (e.g., Core package, Campaign package, etc.). 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. **Table 3-1** lists the packages used throughout the STIX data model specification documents, along with the prefix notation and an example.

Table 3-1. Package prefixes used by the STIX[TM] Language

Package	STIX Core
Prefix	**stix**
Description	The STIX Core data model defines a STIX Package that encompasses all other objects of STIX.
Example	`stix:TTPsType`
Package	**STIX Common**
Prefix	**stixCommon**
Description	The STIX Common data model defines classes that are shared across the various STIX data models.
Example	`stixCommon:ConfidenceType`
Package	**STIX Data Marking**
Prefix	**marking**
Description	The STIX Data Marking data model enables data markings to be used.
Example	`marking:MarkingType`
Package	**STIX Default Vocabularies**
Prefix	**stixVocabs**
Description	The STIX default vocabularies define the classes for default controlled vocabularies used within STIX.
Example	`stixVocabs:MalwareTypeVocab`
Package	**Packages used in STIX Default Extensions**
Prefix	**a (ciq address); capec; ciq; stix-ciqidentity; maec; tlpMarking; cvrf; ioc; oval-def; oval-var**
Description	Various packages are used by STIX extensions. Details are given in *STIX Version 1.2.1 Part 12: Default Extensions*.

Example	`capec:Attack_PatternType`

Package	STIX Basic Data Types
Prefix	**basicDataTypes**
Description	The STIX Basic Data Types data model defines the types used within STIX.
Example	`basicDataTypes:URI`

Package	STIX Indicator
Prefix	**indicator**
Description	The STIX Indicator data model conveys specific Observable patterns combined with contextual information intended to represent artifacts and/or behaviors of interest within a cyber security context.
Example	`indicator:TestMechanismType`

Package	STIX Incident
Prefix	**incident**
Description	The STIX Incident data model captures discrete instances of a specific set of observed events or properties affecting an organization.
Example	`incident:AffectedAssetType`

Package	STIX TTP
Prefix	**ttp**
Description	The STIX TTP data model captures the behavior or modus operandi of cyber adversaries.
Example	`ttp:AttackPatternType`

Package	STIX Campaign
Prefix	**campaign**
Description	The STIX Campaign data model encompasses one or more Threat Actors pursuing an Intended Effect as observed through sets of Incidents and/or TTP, potentially across organizations.
Example	`campaign:AttributionType`

Package	STIX Threat Actor
Prefix	**ta**
Description	The STIX Threat Actor data model captures characterizations of malicious actors (or adversaries) representing a cyber attack threat including presumed intent and historically

	observed behavior.
Example	`ta:ObservedTTPsType`

Package	STIX Exploit Target
Prefix	**et**
Description	The STIX Exploit Target data model conveys a vulnerability or weakness in software, systems, networks or configurations that is targeted for exploitation by the TTP of a Threat Actor.
Example	`et:ConfigurationType`

Package	STIX Course of Action
Prefix	**coa**
Description	The STIX Course of Action data model 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.
Example	`coa:StructuredCOAType`

Package	STIX Report
Prefix	**report**
Description	The STIX Report defines a contextual wrapper for a grouping of STIX content, which could include content specified using any of the other eight top-level constructs, or even other related Reports.
Example	`report:RelatedReportsType`

	CybOX Core
Prefix	**cybox**
Description	The CybOX core data model defines the core constructs used in CybOX.
Example	`cybox:ObservablesType`

3.2 Naming Conventions

The UML classes, enumerations, and properties defined in STIX follow the particular naming conventions outlined in **Table 3-2**.

Table 3-2. Naming formats of different object types

Object Type	Format	Example
Class	CamelCase ending with “Type”	IndicatorBaseType
Property (simple)	Lowercase with underscores between words	capec_id
Property (complex)	Capitalized with underscores between words	Associated_Actor
Enumeration	CamelCase ending with “Enum” or “Type	DateTimePrecisionEnum; IndicatorVersionType
Enumeration value	*varies*	Flash drive; Public Disclosure; Externally-Located
Data type	CamelCase or if the words are acroynms, all capitalized with underscores between words.	PositiveInteger; CVE_ID

3.3 UML Diagrams

This document indicates how UML diagrams are used to visually depict relationships between STIX Language constructs in the rest of the specification. Note that the example diagrams have been extracted directly from the full UML model for STIX; they have not been constructed purely for inclusion in this or the other specification documents. Typically, diagrams are included where the visualization of their relationships between classes is useful for illustration purposes. 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. All component data models include a top-level diagram (see **Figure 3-1**).

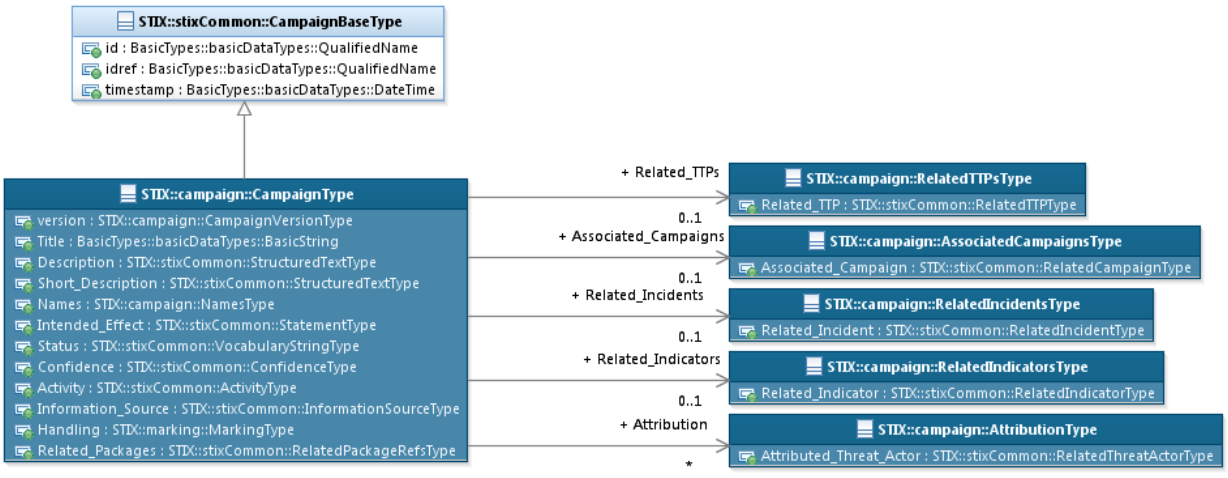

Figure 3-1. Top-level package diagram (Campaign data model)

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 (see **Figure 3-2**).

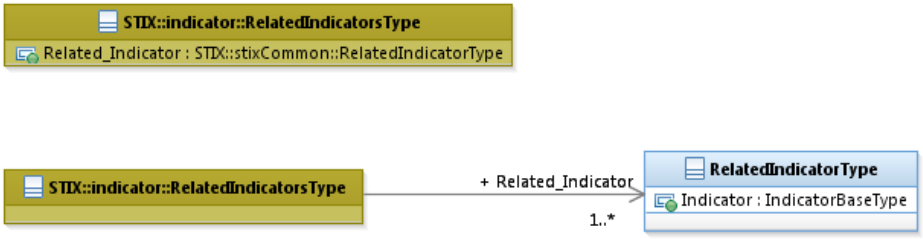

Figure 3-2. Different presentations of class attributes

3.3.1 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.

3.3.2 Diagram Icons and Arrow Types

Diagram icons are used in a UML diagram to indicate whether a shape is a class, enumeration or 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 3-3**.

Table 3-3. 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.

	This arrow type indicates a generalization relationship.

3.3.3 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 collection of specification documents via exemplars are illustrated in **Figure 3-3**.

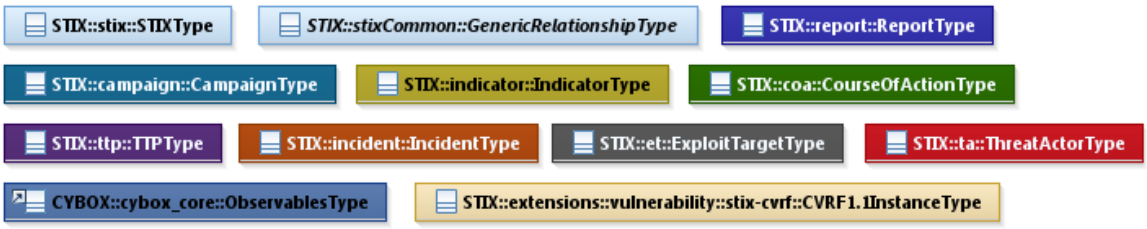

Figure 3-3. Data model color coding

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 Sîrghi, 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 like to thank the STIX Community for its input and help in reviewing this document.

Appendix B. Revision History

Revision	Date	Editors	Changes Made
wd01	11 September 2015	Sean Barnum Desiree Beck Aharon Chernin Rich Piazza	Initial authored draft

Notes

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