

THE MITRE CORPORATION

STIX™ 1.1.1

INCIDENT SPECIFICATION (v1.1.1)

MAY 11, 2015

The Structured Threat Information eXpression (STIX™) framework defines eight 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 Incident construct, which captures discrete instances of a specific set of observed events or properties affecting an organization.

Acknowledgements

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

Trademark Information

STIX, the STIX logo, and CybOX are trademarks of The MITRE Corporation. All other trademarks are the property of their respective owners.

Warnings

MITRE PROVIDES STIX "AS IS" AND MAKES NO WARRANTY, EXPRESS OR IMPLIED, AS TO THE ACCURACY, CAPABILITY, EFFICIENCY, MERCHANTABILITY, OR FUNCTIONING OF STIX. IN NO EVENT WILL MITRE BE LIABLE FOR ANY GENERAL, CONSEQUENTIAL, INDIRECT, INCIDENTAL, EXEMPLARY, OR SPECIAL DAMAGES, RELATED TO STIX OR ANY DERIVATIVE THEREOF, WHETHER SUCH CLAIM IS BASED ON WARRANTY, CONTRACT, OR TORT, EVEN IF MITRE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.¹

Feedback

The STIX development team welcomes any feedback regarding this document. Please send comments, questions, or suggestions to stix@mitre.org.²

¹ For detailed information see [TOU].

² For more information about the STIX Language, please visit [STIX].

Table of Contents

1	Introduction	1
1.1	STIX Specification Documents	1
1.2	Document Conventions	2
1.2.1	Key Words	2
1.2.2	Fonts	2
1.2.3	UML Package References	3
1.2.4	UML Diagrams	3
1.2.4.1	Class Properties	3
1.2.4.2	Diagram Icons and Arrow Types	3
1.2.4.3	Color Coding	4
1.2.5	Property Table Notation	4
1.2.6	Property and Class Descriptions	5
2	Background Information	7
2.1	Incident-Related Component Data Models	7
3	STIX Incident Data Model	10
3.1	IncidentVersionType Enumeration	16
3.2	ExternalIDType Class	17
3.3	TimeType Class	17
3.4	CategoriesType Class	18
3.5	AffectedAssetsType Class	19
3.5.1	AffectedAssetType Class	20
3.5.1.1	AssetTypeType Class	23
3.5.2	NatureOfSecurityEffectType Class	23
3.5.2.1	PropertyAffectedType Class	23
3.6	ImpactAssessmentType Class	27
3.6.1	DirectImpactSummaryType Class	29
3.6.2	IndirectImpactSummaryType Class	31
3.6.3	TotalLossEstimationType Class	33
3.6.3.1	LossEstimationType Class	33
3.6.4	EffectsType Class	33
3.6.5	ExternalImpactAssessmentModelType Class	34
3.7	RelatedIndicatorsType Class	35
3.8	RelatedObservablesType Class	36
3.9	LeveragedTTPsType Class	38
3.10	AttributedThreatActorsType Class	39
3.11	RelatedIncidentsType Class	40
3.12	COATakenType Class and COARequestedType Class	42
3.12.1	ContributorsType Class	43
3.12.2	COATimeType Class	44
3.13	HistoryType Class	44
3.13.1	HistoryItemType Class	46

3.13.1.1 JournalEntryType Class 46

References 47

1 Introduction

The Structured Threat Information eXpression (STIX™) framework defines eight top-level component data models: Observable³, Indicator, Incident, TTP, ExploitTarget, CourseOfAction, Campaign, and ThreatActor. This document serves as the specification for the STIX Incident Version 1.1.1 data model.

As defined within the STIX language, an Incident construct captures discrete instances of a specific set of observed events or properties affecting an organization. More specifically, an Incident consists of properties such as observables, the parties involved, assets affected, impact assessment, leveraged TTPs, attributed threat actors, intended effects, nature of compromise, courses of action requested or taken, confidence in characterization, handling guidance, log of actions taken and source information.

In Section 1.1 we discuss STIX specification documents, and in Section 1.2 we give document conventions. In Section 2, we give background information necessary to fully understand the Incident data model, and we present the Incident data model specification details in Section 3. References are provided in the final section.

1.1 STIX 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 specification overview document provides a comprehensive overview of the full set of STIX data models [STIX_O], which in addition to the eight 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_O] 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 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 (default vocabularies, data marking and extensions), and the color white indicates the component data models. The Observable component data model is shown as an oval shape to indicate that it is defined as a CybOX specification (see [STIX_O] for details). This Incident specification document is highlighted in its associated color (see Section 1.2.4.3). For a list of all STIX documents and related information sources, please see [STIX_O].

³ The CybOX Observable data model is actually defined in the CybOX Language, not in STIX.

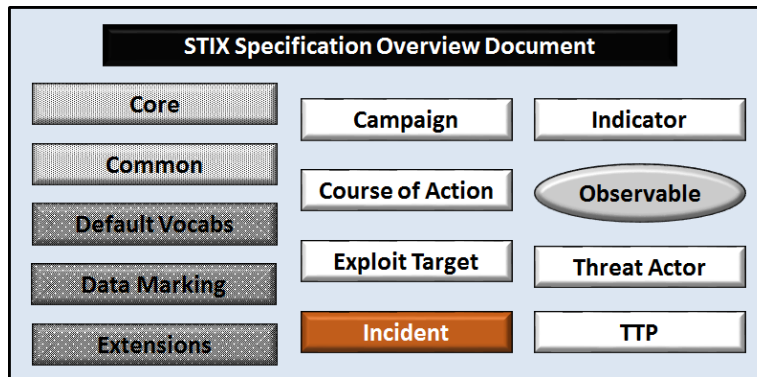


Figure 1-1. STIX Language v1.1.1 specification documents

All specification documents can be found on this STIX Website [STIX-SPECS].

1.2 Document Conventions

The following conventions are used in this document.

1.2.1 Key Words

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 *RFC 2119* [RFC2119].

1.2.2 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 the STIX Specification Overview [STIX₀].

Examples: Incident, Course of Action, Threat Actor

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

Examples: RelatedIncidentsType, 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.*

1.2.3 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. The STIX™ 1.1.1 Specification Overview document [STIX₀] contains a list of the packages used by the Incident 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 Incident data model.

1.2.4 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.

1.2.4.1 Class Properties






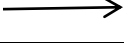

Generally, a class property can be shown in a UML diagram as either an attribute or an association (i.e., the distinction between attributes and associations is somewhat subjective). In order to make the size of UML diagrams in the specifications manageable, we have chosen to capture most properties 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.

1.2.4.2 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 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.
	This arrow type indicates a generalization relationship.

1.2.4.3 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 Incident specification are illustrated via exemplars in Figure 1-2.



Figure 1-2. Data model color coding

1.2.5 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 Indicator data model (see Section 1.2.3).

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.

In addition, properties that are part of a “choice” relationship (e.g., Prop1 OR Prop2 is used but not both) will be denoted by a unique letter subscript (e.g., API_Call_A, Code_B) and single logic expression in the Multiplicity column. For example, if there is a choice of property API_Call_A and Code_B, the expression “A(1)|B(0..1)” will indicate that the API_Call property can be chosen with multiplicity 1 or the Code property can be chosen with multiplicity 0 or 1.

1.2.6 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 Indicator, we write, “The id property specifies a globally unique identifier for the kill chain 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.
	<p><i>Examples:</i></p> <p>The Source 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.</p> <p>The Description property <u>captures</u> a textual description of the Indicator.</p>

<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>

2 Background Information

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

2.1 Incident-Related Component Data Models

As will be explicitly detailed in Section 3, a STIX Incident leverages five other top-level STIX constructs, namely Course of Action, Indicator, Threat Actor, Observable (as defined with the CybOX Language [CybOX_{COR}]) and TTP (as indicated by the outward-oriented arrows). As stated in Section 1.1, each of these components is defined in a separate specification document. Figure 2-1 illustrates the relationship between the Incident and the other core constructs.

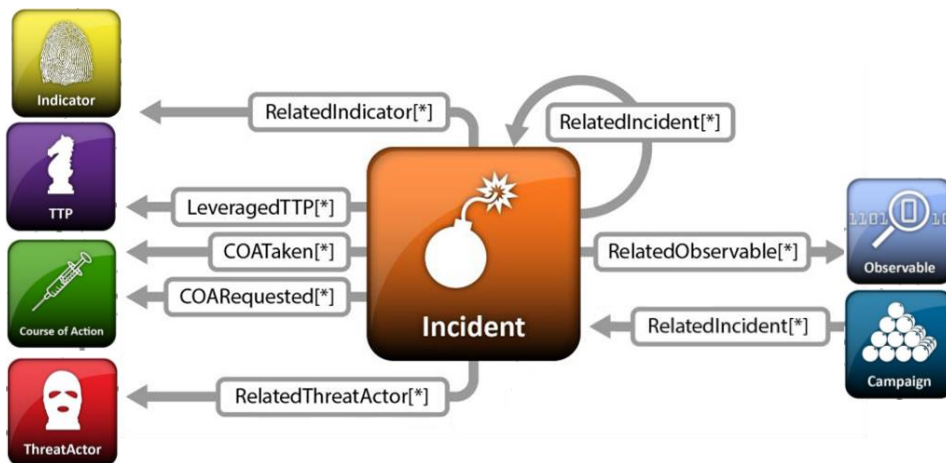


Figure 2-1. Highlevel view of the Incident data model

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

- **Course of Action**

A STIX Course of Action (COA) is used to convey information about courses of action that may be taken either in response to an attack or as a preventative measure prior to an attack. A Course of Action component captures a variety of information such as the Course of Action's objective, likely impact, efficacy, and cost. Please see the STIX Course of Action data model specification [STIX_{COA}] for details.

The Incident data model references the Course of Action data model as a means to identify Courses of Action requested by the incident responders or Courses of Action that were actually taken in the process of responding to the Incident.

- **Incident**

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

- **Indicator**

A STIX Indicator conveys specific Observable patterns combined with contextual information intended to represent artifacts and/or behaviors of interest within a cyber security context. Please see the STIX Indicator data model specification [STIX_{IND}] for details.

The Incident data model leverages the Indicator data model to specify indicators relevant to the Incident whether they were the triggers that initiated the incident response or they are a result of the incident investigation analysis and may be of value in detecting the adversary TTPs leveraged in the incident.

- **Observable**

A STIX Observable (as defined with the CybOX Language⁴) 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 Incident data model leverages the Observable data model to specify the observable instances that were observed in relation to the Incident.

- **Tactics, Techniques and Procedures (TTP)**

A STIX Tactics, Techniques, and Procedures (TTP) is used to represent the behavior or modus operandi of cyber adversaries. Please see the STIX TTP data model specification [STIX_{TTP}] for details.

The Incident data model references the TTP data model as a means to identify sets of specific TTPs that are asserted as having been leveraged in the Incident.

⁴ CybOX specification documents will be created after STIX specification documents are completed.

- **Threat Actor**

A STIX Threat Actor is a characterization of a malicious actor (i.e., adversary) that represents a cyber attack threat. A variety of information can be captured in a Threat Actor construct, including identity, motivations, intended effect, and sophistication level. Please see the STIX Threat Actor data model specification [STIX_{TA}] for details.

The Incident data model references the Threat Actor data model as a means to characterize the Threat Actors that have been attributed to the Incident.

3 STIX Incident Data Model

The primary class of the STIX Incident package is the `IncidentType` class, which characterizes a cyber threat Incident made up of sets of related security events affecting an organization, investigatory details of timing and personnel, as well as other characterizing information discovered or decisions reached during an incident response investigation. Similar to the primary classes of all of the component data models in STIX, the `IncidentType` class extends a base class defined in the STIX Common data model; more specifically, it specializes the `IncidentBaseType` base class, which provides the essential identifier (`id`) and identifier reference (`idref`) properties.

The relationship between the `IncidentType` class and the `IncidentBaseType` base class, as well as the properties of the `IncidentType` class, are illustrated in the UML diagram given in Figure 3-1.

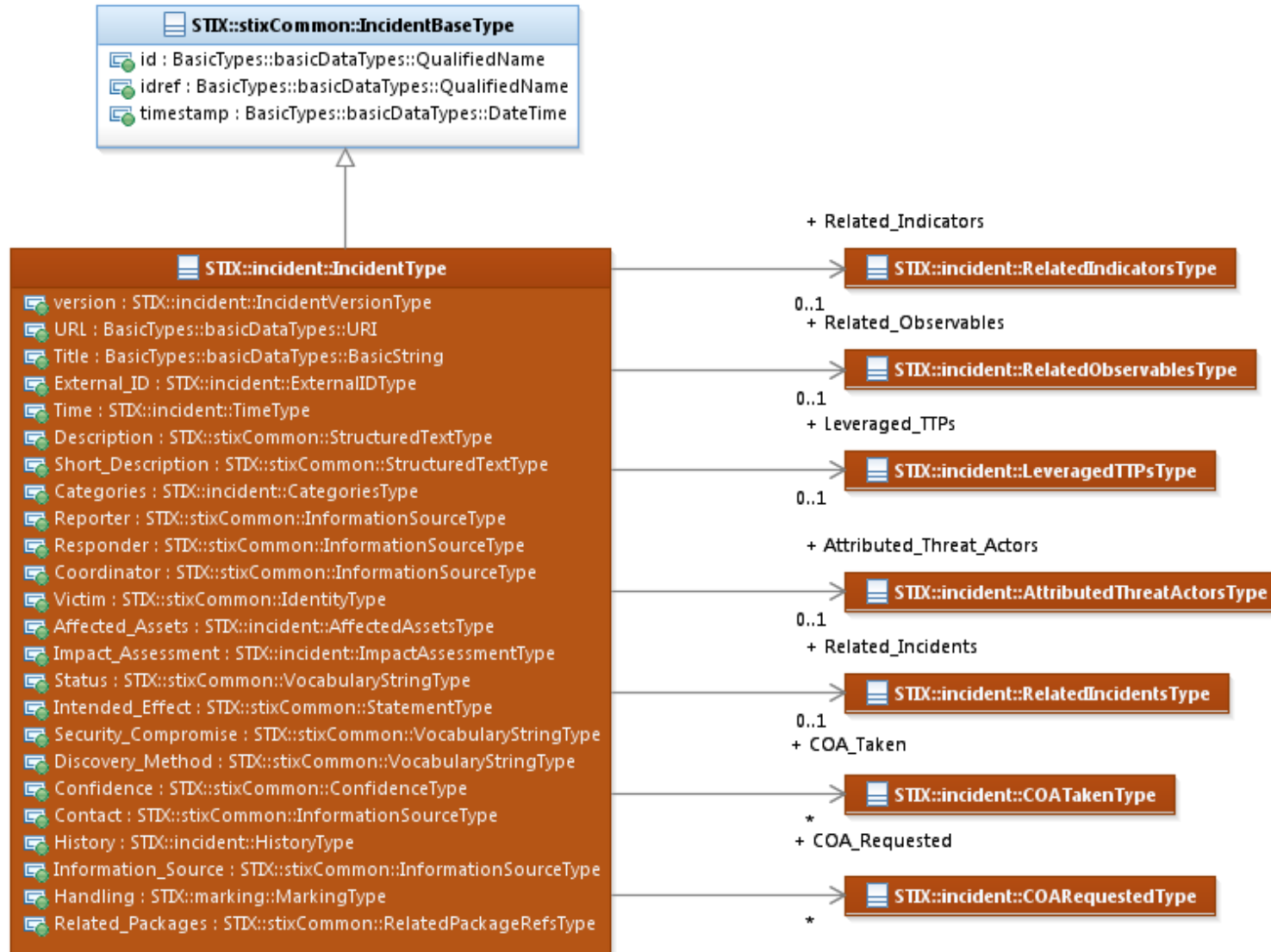


Figure 3-1. UML diagram of the IncidentType class

The property table, which includes property descriptions and corresponds to the UML diagram above, is given in Table 3-1.

All classes defined in the Incident data model are described in detail in Sections 3.1 through 3.13. Details are not provided for classes defined in non-Incident 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 `IncidentType` class

Name	Type	Multiplicity	Documentation
version	<code>IncidentVersionType</code>	0..1	The <code>version</code> property specifies the version identifier of the STIX Incident data model used to capture the information associated with the Incident.
URL	<code>basicDataTypes:URI</code>	0..1	The <code>URL</code> property specifies a URL referencing the location for an external representation of the Incident (e.g. in an incident tracking system).
Title	<code>basicDataTypes:BasicString</code>	0..1	The <code>Title</code> property provides a simple title for the Incident and reflects what the producer thinks the Incident as a whole should be called. Titles are typically used by humans to reference a particular Incident; however, titles are not meant to be used for correlation.
External_ID	<code>ExternalIDType</code>	0..*	The <code>External_ID</code> property captures an identifier for the Incident managed in an external non-STIX system.
Time	<code>TimeType</code>	0..1	The <code>Time</code> property specifies a variety of time values associated with the Incident (e.g., the time the Incident was officially opened).
Description	<code>stixCommon:StructuredTextType</code>	0..1	The <code>Description</code> property captures a textual description of the Incident. 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..1	The <code>Short_Description</code> property captures a short textual description of the Incident. This property is secondary and should only be used if the <code>Description</code> property is already populated and another, shorter description is available.

Categories	CategoriesType	0..1	The Categories property specifies a set of categorization labels for the Incident.
Reporter	stixCommon: InformationSourceType	0..1	The Reporter property characterizes the entity that reported the Incident.
Responder	stixCommon: InformationSourceType	0..*	The Responder property characterizes the entity playing the role of the responder for the Incident.
Coordinator	stixCommon: InformationSourceType	0..*	The Coordinator property characterizes the entity playing the role of coordinator for the Incident.
Victim	stixCommon: IdentityType	0..*	The Victim property characterizes information about a victim of the Incident. For situations calling for more than a simple name, the underlying class may be extended using a more complete structure such as the CIQIdentity3.0InstanceType subclass as defined in the “STIX Extensions Specification Version 1.1.1” document [STIX _{EXT}].
Affected_Assets	AffectedAssetsType	0..1	The Affected_Assets property characterizes the assets affected during the Incident.
Impact_Assessment	ImpactAssessmentType	0..1	The Impact_Assessment property characterizes an assessment of impact for the Incident.
Status	stixCommon: VocabularyStringType	0..1	The Status property specifies the state or disposition of the Incident. Examples of potential statuses are <i>new</i> , <i>open</i> , and <i>closed</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 stixCommon:ControlledVocabularyStringType class. The STIX default vocabulary class for use in the Status property is <i>'IncidentStatusVocab-1.0'</i> .
Related_Indicators	RelatedIndicatorsType	0..1	The Related_Indicators property specifies a set of one or more other Indicators relevant to the Incident whether they were the triggers that initiated the incident response or they are a result of the incident investigation analysis and may be of value in detecting the adversary TTPs leveraged in the incident.

Related_Observables	<code>RelatedObservablesType</code>	0..1	The <code>Related_Observables</code> property specifies a set of one or more observable instances that were observed in relation to the Incident.
Leveraged_TTPs	<code>LeveragedTTPsType</code>	0..1	The <code>Leveraged_TTPs</code> property specifies a set of one or more TTPs that are asserted as having been leveraged in the Incident.
Attributed_Threat_Actors	<code>AttributedThreatActorsType</code>	0..1	The <code>Attributed_Threat_Actors</code> property specifies a set of one or more other Threat Actors that have been attributed to the Incident.
Intended_Effect	<code>stixCommon: StatementType</code>	0..*	The <code>Intended_Effect</code> property characterizes the suspected intended effect of the Incident, which includes a <code>Value</code> property that specifies the type of the effect. Examples of potential types include <i>theft</i> , <i>disruption</i> , and <i>unauthorized access</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 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 <code>Value</code> property is ' <i>IntendedEffectVocab-1.0</i> ' (which is different than the default vocabulary provided for the <code>StatementType</code> class).

Security_Compromise	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Security_Compromise</code> property specifies an assertion of whether the Incident involved a compromise of security properties (e.g. confidentiality). Examples of potential assertions are <i>yes</i> , <i>no</i> , and <i>suspected</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>'SecurityCompromiseVocab-1.0'</i> .
Discovery_Method	<code>stixCommon: VocabularyStringType</code>	0..*	The <code>Discovery_Method</code> property specifies the method by which the Incident was discovered. Examples of potential methods are <i>audit</i> , <i>NIDS</i> , and <i>user</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>'DiscoveryMethodVocab-1.0'</i> .
Related_Incidents	<code>RelatedIncidentsType</code>	0..1	The <code>Related_Incidents</code> property specifies a set of one or more Incidents related to this Incident.
COA_Requested	<code>COARequestedType</code>	0..*	The <code>COA_Requested</code> property specifies one or more Courses of Action for the Incident requested by the incident responders. This property is distinct from the <code>COA_Taken</code> property due to the fact that while incident responders often have rich context for requesting particular courses of action, the authority to actually implement a course of action typically lies with other parties.

COA_Taken	COATakenType	0..*	The <code>COA_Taken</code> property specifies a Course of Action taken for the Incident. This property is distinct from the <code>COA_Requested</code> property due to the fact that while incident responders often have rich context for requesting particular courses of action, the authority to actually implement a course of action typically lies with other parties.
Confidence	stixCommon: ConfidenceType	0..1	The <code>Confidence</code> property characterizes the level of confidence in the accuracy of the overall content captured in the Incident.
Contact	stixCommon: InformationSourceType	0..*	The <code>Contact</code> property characterizes a point of contact for the organizations and personnel involved in the Incident.
History	HistoryType	0..1	The <code>History</code> property captures a log of events or actions taken during the handling of the Incident.
Information_Source	stixCommon: InformationSourceType	0..1	The <code>Information_Source</code> property characterizes the source of the Incident information and all of its contained information. Examples of details captured include identifying characteristics, time-related attributes, and a list of tools used to collect the information.
Handling	marking:MarkingType	0..1	The <code>Handling</code> property specifies the appropriate data handling markings for the properties of this Incident. The marking scope is limited to the Incident and the content it contains. Note that data handling markings can also be specified at a higher level.
Related_Packages	stixCommon: RelatedPackageRefsType	0..1	The <code>Related_Packages</code> property specifies a set of one or more STIX Packages that are related to the Incident.

3.1 IncidentVersionType Enumeration

The `IncidentVersionType` enumeration is an inventory of all versions of the Incident data model that are valid in STIX Version 1.1.1. The enumeration literals are given in Table 3-2.

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

Enumeration Literal	Description
1.0	Incident data model Version 1.0
1.0.1	Incident data model Version 1.0.1
1.1	Incident data model Version 1.1
1.1.1	Incident data model Version 1.1.1

3.2 ExternalIDType Class

The `ExternalIDType` provides a reference to an ID of an incident in a remote system.

The properties of the `ExternalIDType` class are given in Table 3-3.

Table 3-3. Properties of the `ExternalIDType` class

Name	Type	Multiplicity	Description
source	<code>basicDateTypes:</code> <code>NoEmbeddedQuotesString</code>	0..1	The <code>source</code> property specifies the source of the External ID.

3.3 TimeType Class

The `TimeType` class characterizes key time points of interest for the Incident.

The properties of the `TimeType` class are given in Table 3-4.

As specified in [STIX_{COM}], all timestamps specified using the `stixCommon:DateTimeWithPrecisionType` SHOULD include a specification of the time zone. In addition to specifying a date and time, the `Date_Time` property may also capture a

`precision` property to specify the granularity with which the time should be considered, as specified by the `DateTimePrecisionEnum` enumeration (e.g., `'hour'`, `'minute'`). If omitted, the default precision is `'second'`. Digits in a timestamp that are beyond the specified precision SHOULD be zeroed out.

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

Name	Type	Multiplicity	Description
First_Malicious_Action	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>First_Malicious_Action</code> property specifies the time that the first malicious action related to the Incident occurred.
Initial_Compromise	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Initial_Compromise</code> property specifies the time that the initial compromise occurred for the Incident.
First_Data_Exfiltration	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>First_Data_Exfiltration</code> property specifies the first time at which non-public data was taken from the victim environment.
Incident_Discovery	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Incident_Discovery</code> property specifies the first time at which the organization learned the Incident had occurred.
Incident_Opened	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Incident_Opened</code> property specifies the time at which the Incident was officially opened.
Containment_Achieved	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Containment_Achieved</code> property specifies the first time at which the Incident is contained (e.g., the “bleeding is stopped”).
Restoration_Achieved	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Restoration_Achieved</code> property specifies the first time at which the incident's assets are restored (e.g., fully functional).
Incident_Reported	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Incident_Reported</code> property specifies the time at which the Incident was reported.
Incident_Closed	<code>stixCommon:</code> <code>DateTimeWithPrecisionType</code>	0..1	The <code>Incident_Closed</code> property specifies the time at which the Incident was officially closed.

3.4 CategoriesType Class

The `CategoriesType` class specifies one or more category labels for the Incident.

The properties of the `TimeType` class are given in Table 3-5.

Table 3-5. Properties of the `CategoriesType` class

Name	Type	Multiplicity	Description
Category	<code>stixCommon:VocabularyStringType</code>	1..*	The <code>Category</code> property specifies a category label for the Incident. Examples of potential categories are <i>denial of service</i> , <i>improper usage</i> , and <i>scan</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>IncidentCategoryVocab-1.0</i> '.

3.5 AffectedAssetsType Class

The `AffectedAssetsType` class specifies a list of one or more assets affected during the Incident.

The UML diagram corresponding to the `AffectedAssetsType` class is shown in Figure 3-2.

**Figure 3-2.** UML diagram of `AffectedAssetsType` class

The property table given in Table 3-6 corresponds to the UML diagram shown in Figure 3-2.

Table 3-6. Properties of the *AffectedAssetsType* class

Name	Type	Multiplicity	Description
Affected_Asset	<i>AffectedAssetType</i>	0..*	The <i>Affected_Asset</i> property characterizes a particular asset affected during the Incident.

3.5.1 AffectedAssetType Class

The *AffectedAssetType* class characterizes various aspects of the asset negatively impacted by the Incident.

The UML diagram corresponding to the *AffectedAssetsType* class is shown in Figure 3-3.

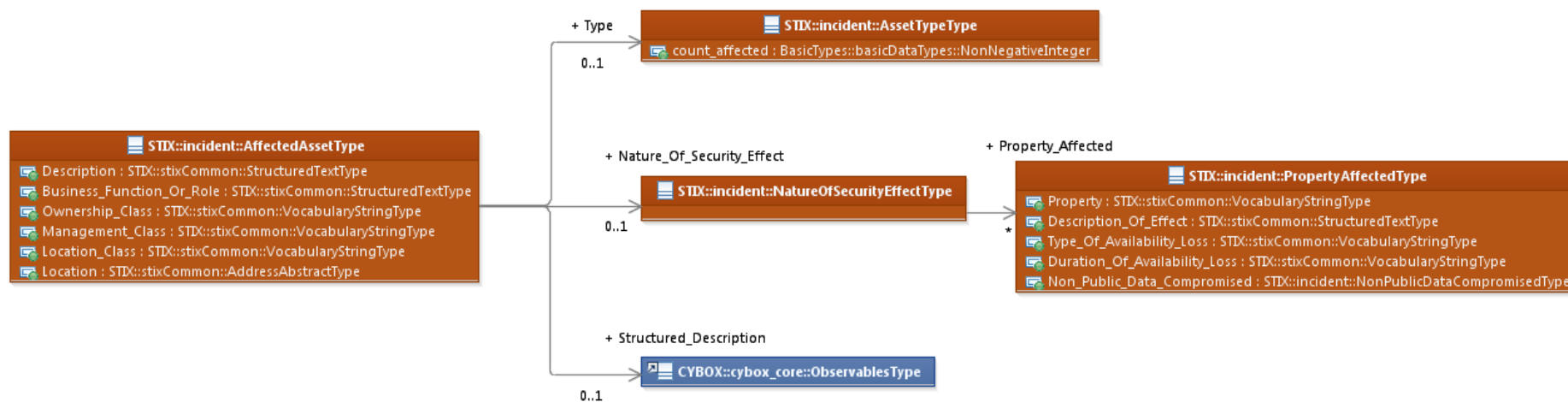


Figure 3-3. UML diagram of *AffectedAssetType* class

The property tables given in Table 3-7, Table 3-8, Table 3-9, Table 3-10 and Table 3-11 all correspond to the UML diagram shown in Figure 3-3.

Table 3-7. Properties of the `AffectedAsset` class

Name	Type	Multiplicity	Description
Type	<code>AssetTypeType</code>	0..1	The <code>Type</code> property characterizes the type of the assets impacted by the Incident.
Description	<code>stixCommon:StructuredTextType</code>	0..1	The <code>Description</code> property captures a textual description of the asset. Any length is permitted. Optional formatting is supported via the <code>structuring_format</code> property of the <code>StructuredTextType</code> class.
Business_Function_Or_Role	<code>stixCommon:StructuredTextType</code>	0..1	The <code>Business_Function_Or_Role</code> captures a textual description of the asset's role, function, and importance within the organization. Any length is permitted. Optional formatting is supported via the <code>structuring_format</code> property of the <code>StructuredTextType</code> class.
Ownership_Class	<code>stixCommon:VocabularyStringType</code>	0..1	The <code>Ownership_Class</code> property specifies who owns (or controls) this asset. Examples of potential values are <i>employee</i> , <i>customer</i> and <i>partner</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>OwnershipClassVocab-1.0</i> '.
Management_Class	<code>stixCommon:VocabularyStringType</code>	0..1	The <code>Management_Class</code> property specifies a high-level characterization of who is responsible for the day-to-day management and administration of the asset. Examples of potential values are <i>internally</i> , <i>externally</i> , and <i>co-managed</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>ManagementClassVocab-1.0.</i> '
Location_Class	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Location_Class</code> property specifies a high-level summarized characterization of the locality type for this asset. Examples of potential values are <i>internal</i> , <i>external</i> , and <i>mobile</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>LocationClassVocab-1.0.</i> '
Location	<code>stixCommon: AddressAbstractType</code>	0..1	The <code>Location</code> property characterizes the actual physical location of the affected asset. The underlying abstract class MUST be extended. The default and strongly RECOMMENDED subclass is <code>CIQAddressInstanceType</code> , as defined in the "STIX Extensions Specification Version 1.1.1" document [STIX _{EXT}].
Nature_Of_Security_Effect	<code>NatureOfSecurityEffectType</code>	0..1	The <code>Nature_Of_Security_Effect</code> property characterizes how the security properties of the asset were affected.
Structured_Description	<code>cybox:ObservablesType</code>	0..1	The <code>Structured_Description</code> property characterizes the asset through specification of a structured cyber Observables instance.

3.5.1.1 AssetTypeType Class

The `AssetTypeType` class characterizes the type of the Affected Asset. Examples of asset types are *directory*, *firewall*, *PBX* and *cashier* (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 `stixCommon:ControlledVocabularyStringType` class. The STIX default vocabulary class for use in this property is *'AssetTypeVocab-1.0'*.

The property of the `AssetTypeType` class is given in Table 3-8.

Table 3-8. Properties of the `AssetTypeType` class

Name	Type	Multiplicity	Description
<code>count_affected</code>	<code>basicDateTypes:Integer</code>	0..1	The <code>count_affected</code> property specifies the number of assets of this type affected in the Incident.

3.5.2 NatureOfSecurityEffectType Class

The `NatureOfSecurityEffectType` class specifies a set of zero or more security properties affected by the Incident.

The property of the `NatureOfSecurityEffectType` class is given in Table 3-9.

Table 3-9. Properties of the `NatureOfSecurityEffectType` class

Name	Type	Multiplicity	Description
<code>Property_Affected</code>	<code>PropertyAffectedType</code>	0..*	The <code>Property_Affected</code> property characterizes how a particular security property of the asset was affected.

3.5.2.1 PropertyAffectedType Class

The `PropertyAffectedType` class characterizes aspects of how security properties of an asset, such as Availability, Confidentiality, etc., were affected in this Incident.

The properties of the `PropertyAffectedType` class are given in Table 3-10.

Table 3-10. Properties of the `PropertyAffectedType` class

Name	Type	Multiplicity	Description
Property	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Property</code> property specifies the security property that was affected by the incident. Examples of potential security properties are <i>confidentiality</i> , <i>integrity</i> and <i>availability</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>LossPropertyVocab-1.0.</i> '
Description_Of_Effect	<code>stixCommon: StructuredTextType</code>	0..1	The <code>Description_Of_Effect</code> property captures a textual description of how the security property was affected. Any length is permitted. Optional formatting is supported via the <code>structuring_format</code> property of the <code>StructuredTextType</code> class.
Type_Of_Availability_Loss	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Type_Of_Availability_Loss</code> property specifies in what manner the availability of the particular asset was affected. Examples of potential values are <i>destruction</i> , <i>deletion</i> and <i>interruption</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>AvailabilityLossTypeVocab-1.1.1</i> '.
Duration_Of_Availability_Loss	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Duration_Of_Availability_Loss</code> property specifies the approximate length of time availability was affected. Examples of potential values are <i>permanent</i> , <i>seconds</i> and <i>days</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>LossDurationVocab-1.0</i> '.
Non_Public_Data_Compromised	<code>NonPublicDataCompromisedType</code>	0..1	The <code>Non_Public_Data_Compromised</code> property specifies whether non-public data was compromised or exposed and whether that data was encrypted or not. Examples of potential values are <i>yes</i> , <i>no</i> and <i>suspected</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 <code>Non_Public_Data_Compromised</code> property is ' <i>SecurityCompromiseVocab-1.0</i> '.

NonPublicDataCompromisedType Class

The `NonPublicDataCompromisedType` class represents whether non-public data was compromised or exposed and whether that data was encrypted or not.

Table 3-11. Properties of the `NonPublicCompromisedType` class

Name	Type	Multiplicity	Description
data_encrypted	<code>basicDateTypes:</code> <code>Boolean</code>	0..1	The <code>data_encrypted</code> property specifies whether the data that was compromised was encrypted or not.

3.6 ImpactAssessmentType Class

The `ImpactAssessmentType` class characterizes a summary assessment of impact for this cyber threat Incident.

The UML diagram corresponding to the `ImpactAssessmentType` class is shown in Figure 3-3.

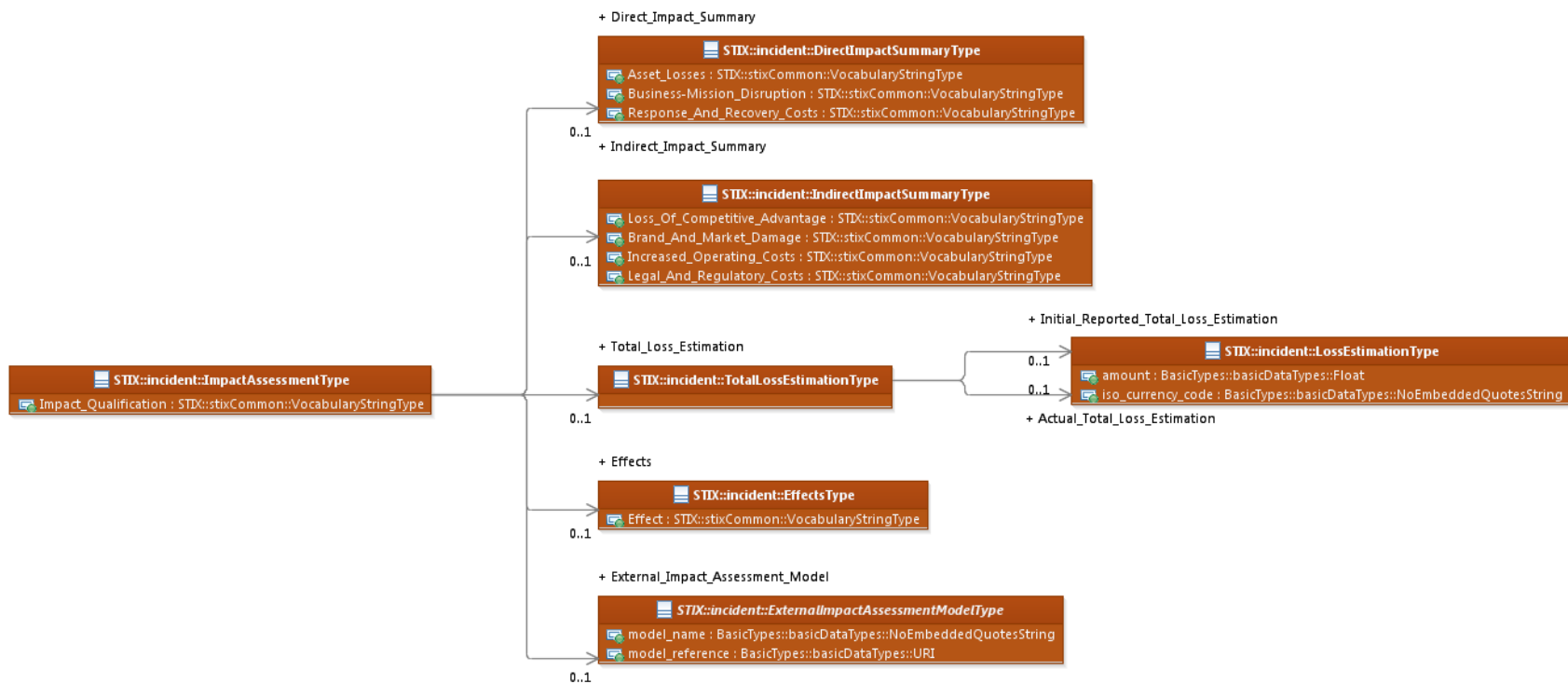


Figure 3-4. UML diagram of the `ImpactAssessmentType` class

The property tables given in Table 3-12, Table 3-13, Table 3-15, Table 3-16, Table 3-17 and Table 3-18 all correspond to the UML diagram shown in Figure 3-4.

Table 3-12. Properties of the `ImpactAssessmentType` class

Name	Type	Multiplicity	Description
Direct_Impact_Summary	<code>DirectImpactSummaryType</code>	0..1	The <code>Direct_Impact_Summary</code> property characterizes (at a high level) impact directly resulting from the Threat Actor's actions against organizational assets within the Incident.
Indirect_Impact_Summary	<code>IndirectImpactSummaryType</code>	0..1	The <code>Indirect_Impact_Summary</code> property characterizes (at a high level) impact from other stakeholder reactions to the Incident.
Total_Loss_Estimation	<code>TotalLossEstimationType</code>	0..1	The <code>Total_Loss_Estimation</code> property specifies the total estimated financial loss for the Incident.
Impact_Qualification	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Impact_Qualification</code> property specifies the subjective level of impact of the Incident. Examples of potential values are <i>insignificant</i> , <i>catastrophic</i> and <i>damaging</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 <code>Impact_Qualification</code> property is ' <i>ImpactQualificationVocab-1.0</i> '.

Effects	EffectsType	0..1	The <code>Effects</code> property specifies a set of one or more effects of this incident.
External_Impact_Assessment_Model	ExternalImpactAssessmentModelType	0..1	The <code>External_Impact_Assessment_Model</code> property characterizes impact assessment details. It is defined utilizing an abstract class enabling the definition through extension of incident impact assessment models external to STIX.

3.6.1 DirectImpactSummaryType Class

The `DirectImpactSummaryType` class quantitatively characterizes (at a high level) the direct impact of the Incident, both financial and non-financial.

Table 3-13. Properties of the `DirectImpactSummaryType` class

Name	Type	Multiplicity	Description
Asset_Losses	stixCommon: VocabularyStringType	0..1	The <code>Asset_Losses</code> property specifies (at a high level) the level of asset-related losses that occurred in the Incident, including lost or damaged assets, stolen funds, cash outlays, etc. Examples of potential levels are <i>minor</i> , <i>major</i> and <i>none</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>'ImpactRatingVocab-1.0'</i> .
Business-Mission_Disruption	stixCommon: VocabularyStringType	0..1	The <code>Business-Mission_Disruption</code> property specifies (at a high level) the level of business or mission disruption impact that occurred in the Incident including unproductive man-hours, lost revenue from system downtime, etc.

			<p>Examples of potential levels are <i>minor</i>, <i>major</i> and <i>none</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 <code>property</code> is <i>'ImpactRatingVocab-1.0'</i>.</p>
Response_And_Recovery_Costs	<code>stixCommon: VocabularyStringType</code>	0..1	<p>The <code>Response_And_Recovery_Costs</code> property specifies (at a high level) the level of response and recovery related costs that occurred in the Incident including cost of response, investigation, remediation, restoration, etc.</p> <p>Examples of potential levels are <i>minor</i>, <i>major</i> and <i>none</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>'ImpactRatingVocab-1.0'</i>.</p>

3.6.2 IndirectImpactSummaryType Class

The `IndirectImpactSummaryType` class qualitatively characterizes (at a high level) the indirect impact of the Incident, both financial and non-financial.

Table 3-14. Properties of the `IndirectImpactSummaryType` class

Name	Type	Multiplicity	Description
Loss_Of_Competitive_Advantage	<code>stixCommon:VocabularyStringType</code>	0..1	The <code>Loss_Of_Competitive_Advantage</code> if a loss of competitive advantage occurred in the Incident. The impact could include: loss/damage/exposure of IP, corporate wisdom, ability to compete, key personnel, etc. Examples of potential statuses are <i>yes</i> , <i>no</i> and <i>suspected</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>'SecurityCompromiseVocab-1.0'</i> .
Brand_And_Market_Damage	<code>stixCommon:VocabularyStringType</code>	0..1	The <code>Brand_And_Market_Damage</code> property specifies the level of impact based on brand or market damage that occurred in the Incident. The impact could include: lost customers or partners, decrease in market value or share, advertising, rebranding, etc. Examples of potential statuses are <i>yes</i> , <i>no</i> and <i>suspected</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 this property is <i>'SecurityCompromiseVocab-1.0'</i> .
Increased_Operating_Costs	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Increased_Operating_Costs</code> property specifies if increased operating costs occurred in the Incident. The impact could include: cost of additional audits, new hires or training, mandatory action, higher insurance, etc. Examples of potential statuses are <i>yes</i> , <i>no</i> and <i>suspected</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 this property is <i>'SecurityCompromiseVocab-1.0'</i> .
Legal_And_Regulatory_Costs	<code>stixCommon: VocabularyStringType</code>	0..1	The <code>Legal_And_Regulatory_Costs</code> property specifies if legal and regulatory costs occurred in the Incident. This includes legal fees, lawsuits, customer damages, contract violations, etc. Examples of potential statuses are <i>yes</i> , <i>no</i> and <i>suspected</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 this property is <i>'SecurityCompromiseVocab-1.0'</i> .

3.6.3 TotalLossEstimationType Class

The `TotalLossEstimationType` class characterizes both the initial reported and actual estimated financial losses for this Incident.

Table 3-15. Properties of the `TotalLossEstimationType` class

Name	Type	Multiplicity	Description
Initial_Reported_Total_Loss_Estimation	<code>LossEstimationType</code>	0..1	The <code>Initial_Reported_Total_Loss_Estimation</code> property specifies the initially reported level of total estimated financial loss for the Incident.
Actual_Total_Loss_Estimation	<code>LossEstimationType</code>	0..1	The <code>Actual_Total_Loss_Estimation</code> property specifies the actual level of total estimated financial loss for the Incident.

3.6.3.1 LossEstimationType Class

The `LossEstimationType` class characterizes an estimated financial loss.

Table 3-16. Properties of the `LossEstimationType` class

Name	Type	Multiplicity	Description
amount	<code>basicDateTypes:Decimal</code> ⁵	0..1	The <code>amount</code> property specifies the estimated financial loss for the Incident.
iso_currency_code	<code>basicDataTypes:NoEmbeddedQuotesString</code> ⁶	0..1	The <code>iso_currency_code</code> property specifies the ISO 4217 currency code if other than USD.

3.6.4 EffectsType Class

The `EffectsType` class specifies one or more effects asserted as present for this Incident.

⁵ The type of the `amount` property is suggested, not normative.

⁶ The type of the `iso_currency_code` property is suggested, not normative.

Table 3-17. Properties of the `EffectsType` class

Name	Type	Multiplicity	Description
Effect	<code>stixCommon:VocabularyStringType</code>	1..*	The <code>Effect</code> property represents a single effect asserted as present for this Incident. Examples of potential statuses are <i>denial of service</i> , <i>improper usage</i> and <i>scan</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 this property is ' <i>IncidentEffectVocab-1.0</i> '.

3.6.5 ExternalImpactAssessmentModelType Class

The `ExternalImpactAssessmentModelType` class is an abstract class enabling the definition through extension of incident impact assessment models external to STIX.

Table 3-18. Properties of the `ExternalImpactAssessmentModelType` class

Name	Type	Multiplicity	Description
model_name	<code>basicDateTypes:NoEmbeddedQuotesString</code>	0..1	The <code>model_name</code> property specifies the name of the externally defined impact assessment model.
model_reference	<code>BasicDataType:URI</code>	0..1	The <code>model_reference</code> property specifies a URI reference to the characterization of the externally defined impact assessment model.

3.7 RelatedIndicatorsType Class

The `RelatedIndicatorsType` class specifies one or more Indicators relevant to the Incident whether they were the triggers that initiated the incident response or they are a result of the incident investigation analysis and may be of value in detecting the adversary TTPs leveraged in the incident. 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 `RelatedIndicatorsType` class is shown in Figure 3-5, and the specialized properties are shown in Table 3-19.

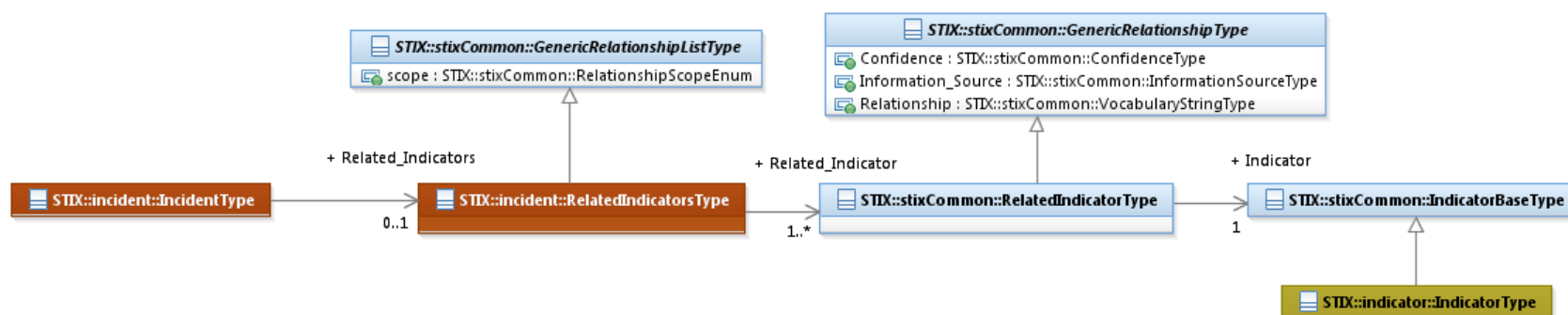


Figure 3-5. UML diagram of the `RelatedIndicatorsType` class

The property table given in Table 3-19 corresponds to the UML diagram given in Figure 3-5.

Table 3-19. Properties of the `RelatedIndicatorsType` class

Name	Type	Multiplicity	Description
Related_Indicator	<code>stixCommon:RelatedIndicatorType</code>	1..*	The <code>Related_Indicator</code> property characterizes an Indicator asserted to be relevant to the Incident whether they were the triggers that initiated the incident response or they are a result of the incident investigation analysis and may be of value in detecting the adversary TTPs leveraged in the incident. To further characterize the relationship to the Indicator,

			information captured includes the level of confidence that the Indicator is relevant, the source of the relationship information, and type of the relationship.
--	--	--	---

3.8 RelatedObservablesType Class

The `RelatedObservablesType` class specifies one or more CybOX Observable instances that were observed in relation to the Incident. 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 `RelatedObservablesType` class is shown in Figure 3-6, and the properties are shown in Table 3-20.

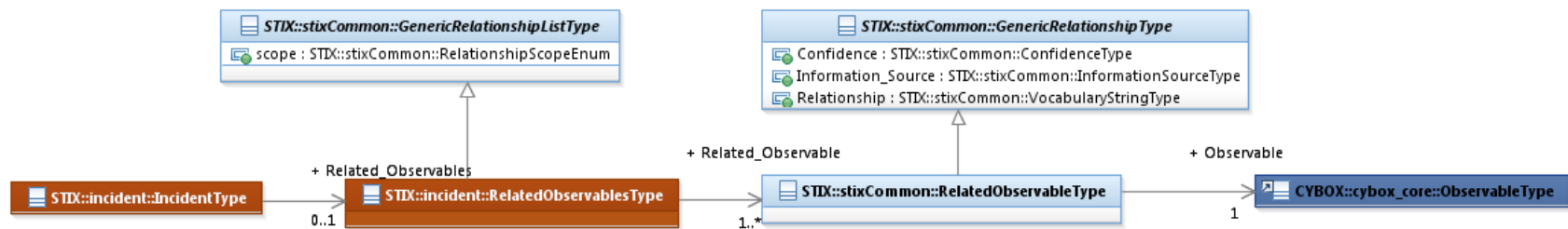


Figure 3-6. UML diagram of the `RelatedObservablesType` class

The property table given in Table 3-20 corresponds to the UML diagram shown in Figure 3-6.

Table 3-20. Properties of the `RelatedObservablesType` class

Name	Type	Multiplicity	Description
Related_Observable	<code>stixCommon: RelatedObservableType</code>	1..*	The <code>Related_Observable</code> property captures the properties of a cyber <code>Observable</code> instance that was observed in relation to the Incident. In addition, the property characterizes the relationship between the <code>Observable</code> and the Incident by capturing additional information such as the level of confidence in the assertion that the <code>Observable</code> and the Incident are related, information on the source of the relationship information, and details on the type of the relationship between the <code>Observable</code> and the Incident.

3.9 LeveragedTTPsType Class

The `LeveragedTTPsType` class specifies one or more TTP that are asserted to have been leveraged during this Incident. 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 `LeveragedTTPsType` class is shown in Figure 3-7.

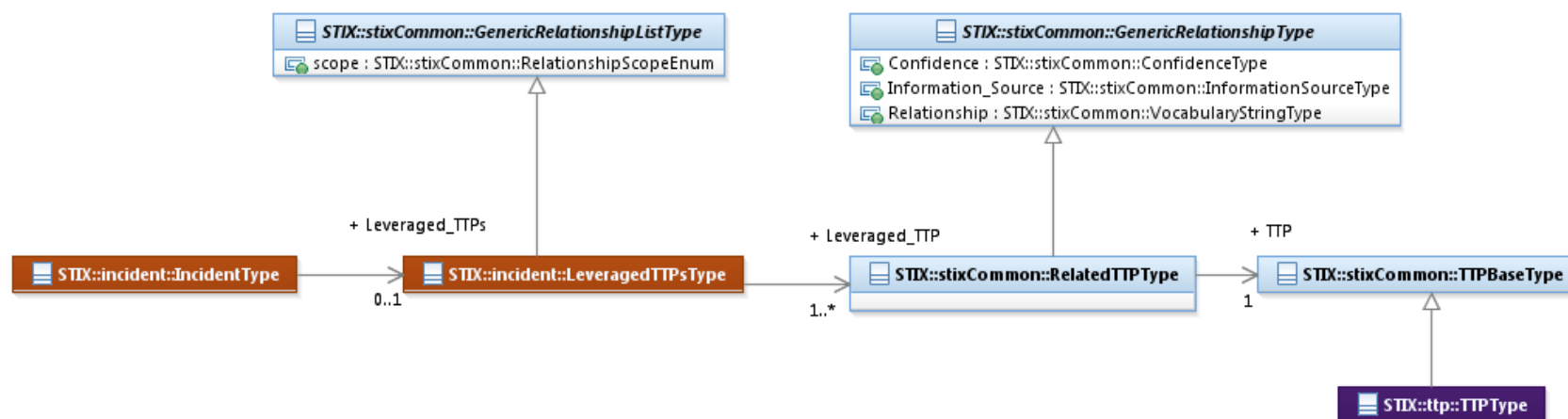


Figure 3-7. UML diagram of the `LeveragedTTPsType` class

The property table given in Table 3-21 corresponds to the UML diagram shown in Figure 3-7.

Table 3-21. Properties of the `LeveragedTTPsType` class

Name	Type	Multiplicity	Description
Leveraged_TTP	<code>stixCommon: RelatedTTPType</code>	1..*	The <code>Leveraged_TTP</code> property specifies a TTP asserted to have been leveraged in the Incident and characterizes the relationship between the Incident and the TTP by capturing information such as the level of confidence that the Incident and the TTP are related, the source of the relationship information, and the type of relationship.

3.10 AttributedThreatActorsType Class

The `AttributedThreatActorsType` class specifies a list of one or more Threat Actors that have been attributed to the Incident. 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 `AttributedThreatActorsType` class is shown in Figure 3-8.

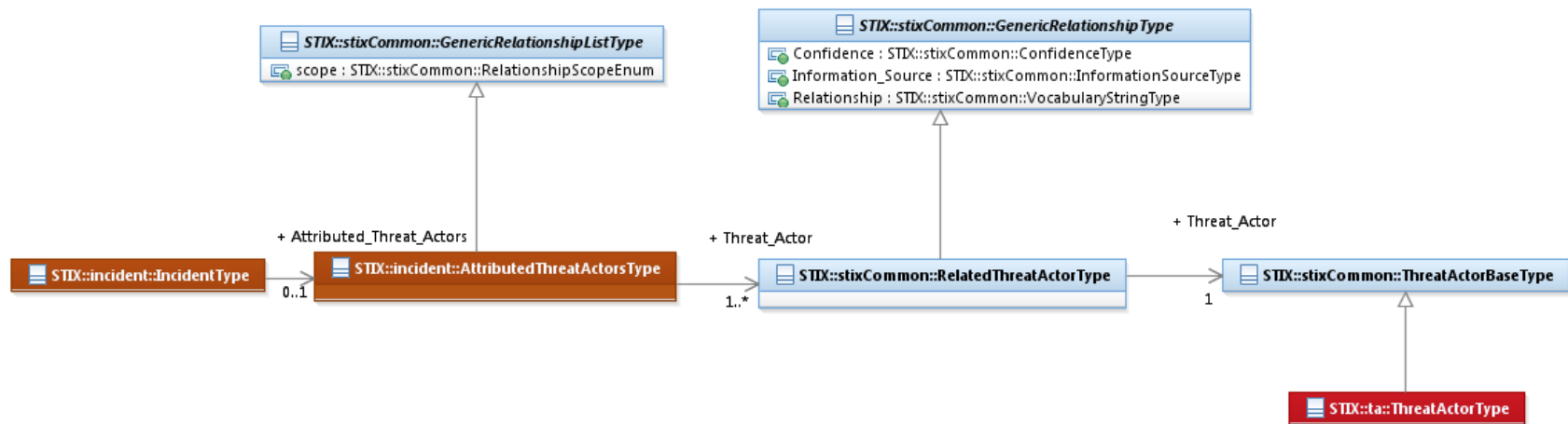


Figure 3-8. UML diagram of the `AttributedThreatActorType` class

The property table given in Table 3-22 corresponds to the UML diagram shown in Figure 3-8.

Table 3-22. Properties of the `AttributedThreatActorsType` class

Name	Type	Multiplicity	Description
Threat_Actor	<code>stixCommon: RelatedThreatActorType</code>	1..*	The <code>Threat_Actor</code> property captures a relationship to a Threat Actor that has been attributed to the Incident. To further characterize the relationship between the Incident and the Threat Actor, information captured includes the level of confidence that the Incident and the Threat Actor are related, the source of the relationship information, and type of the relationship.

3.11 RelatedIncidentsType Class

The `RelatedIncidentsType` class specifies a list of one or more other Incidents asserted as related to the Incident 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).

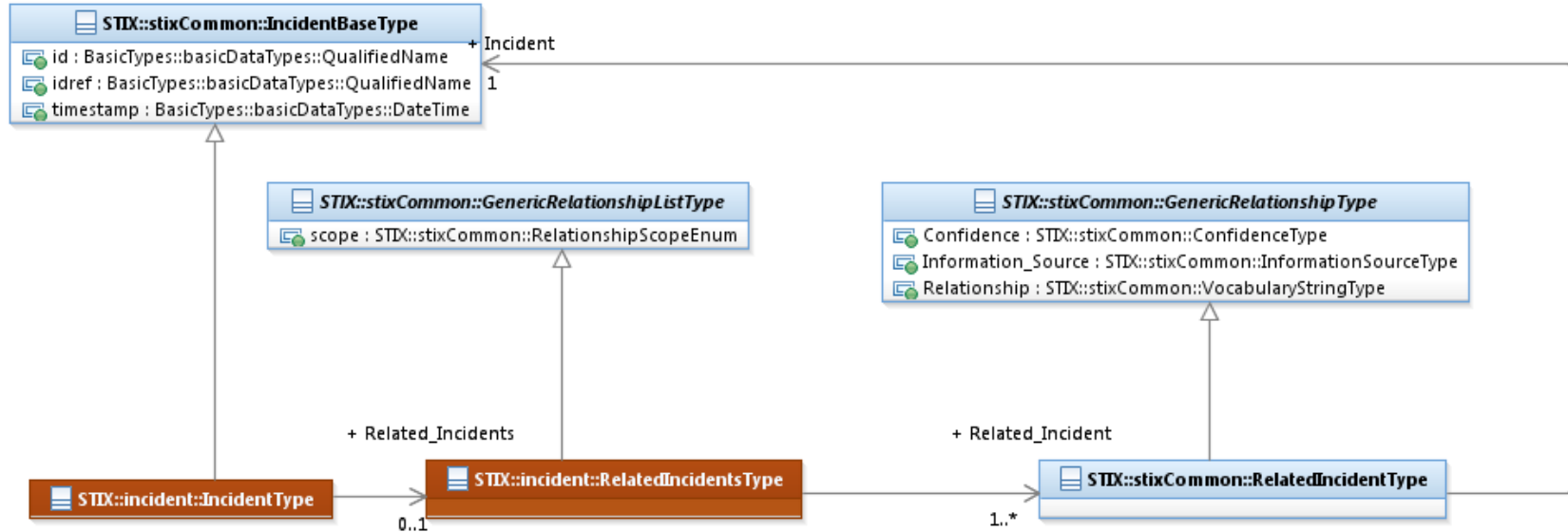


Figure 3-9. UML diagram of the `RelatedIncidentsType` class

The property table given in Table 3-23 corresponds to the UML diagram given in Figure 3-9.

Table 3-23. Properties of the `RelatedIncidentsType` class

Name	Type	Multiplicity	Description
Related_Incident	<code>stixCommon: RelatedIncidentType</code>	1..*	The <code>Related_Incident</code> property specifies another Incident associated with this Incident and characterizes the relationship between the Incidents by capturing information such as the level of confidence that the Incidents are related, the source of the relationship information, and type of the relationship. A relationship between Incidents may represent assertions of general associativity or different versions of the same Incident.

3.12 COATakenType Class and COARequestedType Class

The COATakenType class specifies a Course of Action for the Incident requested by the incident responders. The COARequestedType class specifies a Course of Action taken for the Incident. The UML diagram corresponding to the COATakenType and COARequestedType classes is shown in Figure 3-10.

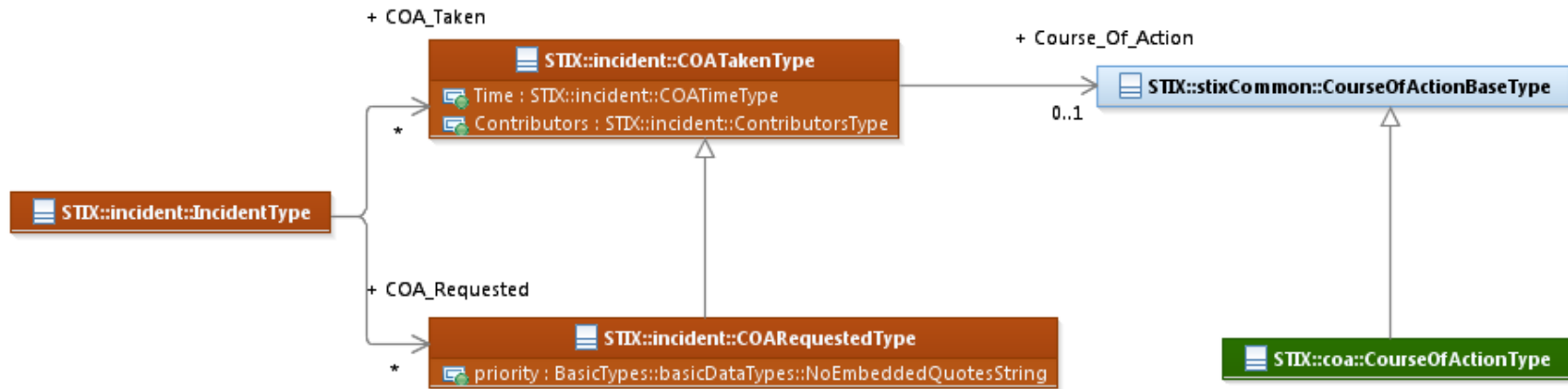


Figure 3-10. UML diagram of the COATakenType and COARequestedType classes

The property tables given in Table 3-24, Table 3-25, Table 3-26 and Table 3-27 all correspond to the UML diagram given in Figure 3-10.

Table 3-24. Properties of the COATakenType class

Name	Type	Multiplicity	Description
Time	COATimeType	0..1	The Time property specifies when this Course of Action was taken (start and end).
Contributors	ContributorsType	0..1	The Contributors property specifies contributing actors for the Course of Action taken.

Course_Of_Action	<code>stixCommon:CourseOfActionBaseType</code>	0..1	The <code>Course_Of_Action</code> property specifies the actual Course of Action taken. If a new Course of Action is defined (as opposed to an existing Course of Action referenced), the default and strongly RECOMMENDED method is to leverage the <code>CourseOfActionType</code> class from the Course Of Action data model (which extends the <code>CourseOfActionBaseType</code> superclass).
-------------------------	--	------	---

Table 3-25. Properties of the `COARequestedType` class

Name	Type	Multiplicity	Description
priority	<code>basicDataTypes:NoEmbeddedQuotesString</code> ⁷	0..1	The <code>priority</code> property characterizes a suggested level of priority to be applied to this requested COA.

3.12.1 ContributorsType Class

The `ContributorType` class characterizes the actors involved in a course of action.

Table 3-26. Properties of the `ContributorsType` class

Name	Type	Multiplicity	Description
Contributor	<code>cyboxCommon:ContributorType</code>	1..*	The <code>Contributor</code> property characterizes an entity involved in this Course of Action.

⁷ The type of the `priority` property is suggested, not normative.

3.12.2 COATimeType Class

The COATimeType class specifies the relevant time period for the execution of a courses of action were for this Incident.

Table 3-27. Properties of the COATimeType class

Name	Type	Multiplicity	Description
Start	stixCommon: DateTimeWithPrecisionType	0..1	The <code>Start</code> property specifies the time in which the Course of Action was begun. To avoid ambiguity, timestamps SHOULD include a specification of the time zone. In addition to capturing a date and time, the <code>Start</code> property MAY also capture a <code>precision</code> property to specify the granularity with which the time should be considered, as specified by the <code>DateTimePrecisionEnum</code> enumeration (e.g., <code>'hour'</code> , <code>'minute'</code>). If the <code>Start</code> property is not present, then it is unknown when the Course of Action was started.
End	stixCommon: DateTimeWithPrecisionType	0..1	The <code>End</code> field specifies the time at which the Course of Action was completed. In order to avoid ambiguity, it is strongly suggest that all timestamps include a specification of the timezone if it is known. In addition to capturing a date and time, the <code>End</code> property MAY also capture a <code>precision</code> property to specify the granularity with which the time should be considered, as specified by the <code>DateTimePrecisionEnum</code> enumeration (e.g., <code>'hour'</code> , <code>'minute'</code>). If the <code>End</code> property is not present, then it is unknown when the Course of Action ended, or the Course of Action is ongoing.

3.13 HistoryType Class

The `HistoryType` class captures a record of events or actions taken as well as information discovered during the handling of the Incident. This can include Courses of Action taken and general journal notes. The time that the note is written, or the Course of Action taken and the author or actors involved may be specified.

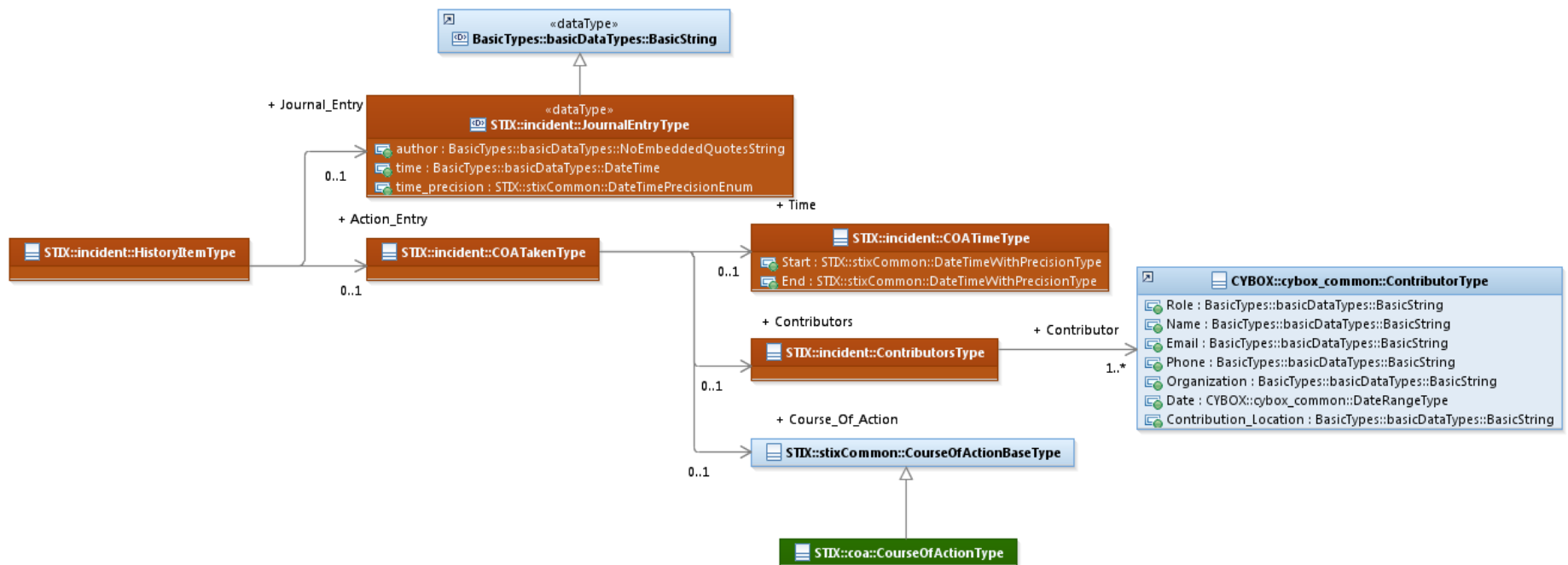


Figure 3-11. UML diagram of the HistoryType class

The property tables given in Table 3-28, Table 3-29 and Table 3-30 corresponds to the UML diagram given in Figure 3-11. Also see tables in Section 3.12.

Table 3-28. Properties of the HistoryType class

Name	Type	Multiplicity	Description
History_Item	HistoryItemType	0..*	The History_Item property captures a log entry of either an event or action taken during the handling of the Incident or a journal entry containing information discovered during the investigation of the Incident.

3.13.1 HistoryItemType Class

The `HistoryItemType` class specifies the choice of either an action or journal entry as an item in the Incident's history.

Table 3-29. Properties of the `HistoryItemType` class

Name	Type	Multiplicity	Description
Action_Entry	<code>COATakenType</code>	0..1	The <code>Action_Entry</code> property captures a record of a Course of Action taken during the handling of the Incident.
Journal_Entry	<code>JournalEntryType</code>	0..1	The <code>Journal_Entry</code> property captures journal notes for information discovered during the handling of the Incident.

3.13.1.1 JournalEntryType Class

The `JournalEntryType` class captures journal notes for information discovered during the handling of the Incident. It is a subtype of `BasicDataTypes:BasicString` (see Figure 3-11).

Table 3-30. Properties of the `JournalEntryType` class

Name	Type	Multiplicity	Description
author	<code>basicDataTypes:NoEmbeddedQuotesString</code>	0..1	The <code>author</code> property specifies the author of the <code>JournalEntry</code> note.
time	<code>BasicDataTypes:DateTime</code>	0..1	The <code>time</code> property specifies the date and time of the journal entry creation. To avoid ambiguity, all timestamps SHOULD include a specification of the time zone.
time_precision	<code>stixCommon:DateTimePrecisionEnum</code>	0..1	The <code>time_precision</code> property specifies the granularity with which the <code>time</code> property should be considered, as specified by the <code>DateTimePrecisionEnum</code> enumeration (e.g., <i>hour</i> , <i>minute</i>). If omitted, the default precision is <i>second</i> . Digits in a timestamp that are beyond the specified precision should be zeroed out.

References

References made in this document are listed below.

- [CybOX_{COR}] CybOX™ Core Specification (*not yet available*).
- [RFC2119] RFC 2119 – Key words for use in RFCs to Indicate Requirement Levels
<http://www.ietf.org/rfc/rfc2119.txt>
- [STIX] STIX™ Web Site
<https://stix.mitre.org>
- [STIX-SPECS] STIX™ Project Github Site
<http://github.com/STIXProject/specifications>
- [STIX_{CAM}] STIX™ 1.1.1 Campaign Specification (v1.1.1)
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [STIX_{COA}] STIX™ 1.1.1 Course of Action Specification (v1.1.1)
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [STIX_{IND}] STIX™ 1.1.1 Indicator Specification (v2.1.1)
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [STIX_O] STIX™ 1.1.1 Specification Overview
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [STIX_{TA}] STIX™ 1.1.1 TTP Specification (v1.1.1)
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [STIX_{TTP}] STIX™ 1.1.1 Threat Actpr Specification (v1.1.1)
<http://stix.mitre.org/about/documents/XXXX.pdf>
- [TOU] Terms of Use
<http://stix.mitre.org/about/termsfuse.html>