THE MITRE CORPORATION

STIX™ 1.1.1
CAMPAIGN SPECIFICATION (V1.1.1)

APRIL 20, 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 Campaign construct, which encompasses one or more Threat Actors pursuing an Intended Effect as observed through sets of Incidents and/or TTP, potentially across organizations.

Acknowledgements

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

Trademark Information

Warnings

Feedback

The STIX development team welcomes any feedback regarding the STIX Campaign Specification. Please send any 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

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

As defined within the STIX language, a Campaign construct is an instance of a Threat Actor (adversary), whether characterized or not, pursuing an Intended Effect as observed through sets of Incidents and/or TTP, potentially across organizations. In addition to Threat Actor, Intended Effect, Incident, and TTP information, a Campaign construct may also comprise a variety of additional information, including status of the Campaign, a textual description, and alias names for the Campaign.

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 Campaign data model, and we present the Campaign data model specification details in Section 3. The appendix gives information about corresponding XML implementations. References are provided in the final section.

1.1 STIX Specification Documents

The STIX specification corresponds to 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 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, 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 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 (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 Campaign 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 [$STIX_O$].

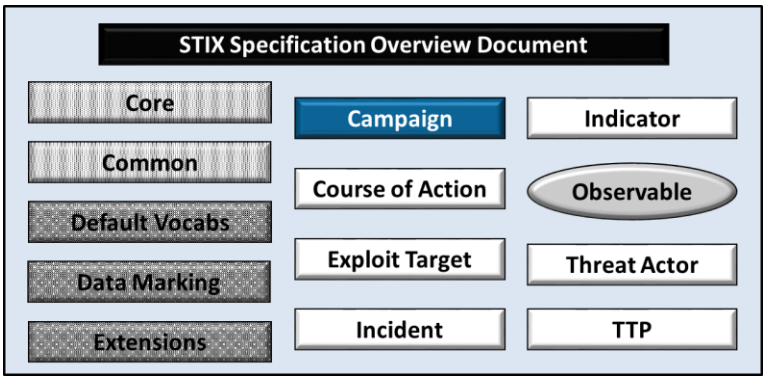

Figure 1-1. STIX Language v1.1.1 specification documents

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

1.2 Document Conventions

The following conventions are used in this document.

1.2.1 Keywords

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 [$STIX_O$].

  Examples: Indicator, Course of Action, Threat Actor

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

  Examples: `RelatedIndicatorsType`, `stixCommon:StatementType`

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

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

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

1.2.3 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 [$STIX_O$] contains a list of the packages used by the Campaign data model, along with the associated prefix notation, a description, and an example.

Note that in this specification document, we do not explicitly specify the package prefix for any classes that originate from the Campaign 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 would be for classes that specialize a superclass, and for 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 other attributes than the ones 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 a data type, and decorative icons are used to indicate whether an element is an attribute of a class or an enumeration literal. In addition, two different arrow styles indicate either a directed association relationship (regular arrowhead) or a generalization relationship (triangle-shaped arrowhead). The icons and arrow styles we use are shown and described in Table 1-1.

Table 1-1. UML diagram icons

Icon	Description
	This diagram icon indicates a class. If the name is in italics, it is an abstract class.
	This diagram icon indicates an enumeration.
	This diagram icon indicates a data type.
	This decorator icon indicates an attribute of a class. The green circle means its visibility is public. If the circle is red or yellow, it means its visibility is private or protected.
	This decorator icon indicates an enumeration literal.
	This arrow type indicates a directed association relationship.
	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 Campaign specification are illustrated 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 Campiagn 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 Campaign, we write, "The `id` property specifies a globally unique identifier for the Campaign instance." In fact, the verb "specifies" could have been replaced by any number of alternatives: "defines," "describes," "contains," "references," etc.

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

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

Verb	STIX Definition
captures	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.
	Examples: The `Source` property characterizes the source of the sighting information. Examples of details captured include identitifying characteristics, time-related attributes, and a list of the tools used to collect the information. The `Description` property captures a textual description of the Indicator.

characterizes	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.
	Example: The `Confidence` property characterizes the level of confidence in the accuracy of the overall content captured in the Incident. The `ActivityType` class characterizes basic information about an activity a defender might use in response to a Campaign.
specifies	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.
	Example: The `version` property specifies the version identifier of the STIX Campaign data model used to capture the information associated with the Campaign.

2 Background Information

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

2.1 Campaign-Related Component Data Models

As will be explicitly detailed in Section 3, a STIX Campaign leverages four other core STIX constructs, namely Threat Actor, TTP, Incident, and Indicator (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 Campaign and the other core constructs.

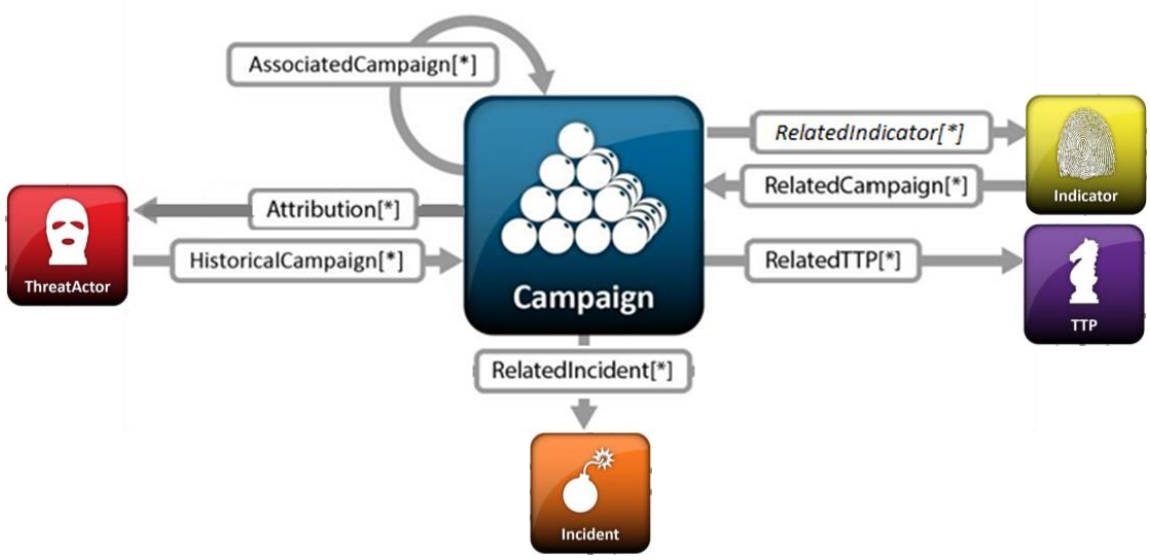

Figure 2-1. High level view of the Campaign data model

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

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

- **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.

Version 1.1.1 of the Campaign data model references the Indicator data model as a means of referring to indicators relevant to the Campaign. Beginning in Campaign Version 1.1, this reference relationship was deprecated; however, it remains in v1.1.1 for backward compatibility. The relationship will be removed in Campaign Version 2.0.

Under Version 1.1.1 – unless backward compatibility is necessary – relationships between Indicators and Campaigns SHOULD be captured from the Indicator to Campaign direction (i.e., Indicators SHOULD reference associated Campaigns rather than the other way around). Figure 2-1 shows the deprecated direction using italics.

- **Incident**
  An STIX Incident corresponds to sets of related security events affecting an organization, along with information discovered or decided during an incident response investigation. Please see the STIX Incident data model specification [$STIX_{INC}$] for details.

  The Campaign data model references the Incident data model in order to identify sets of observed Incidents that are part of the Campaign (or in some way related to the Campaign).

- **Tactics, Techniques and Procedures (TTP)**
  A STIX Tactics, Techniques, and Procedures (TTP) are representations of the behavior or modus operandi of cyber adversaries. Please see the STIX TTP data model specification [$STIX_{TTP}$] for details.

  The Campaign data model references the TTP data model as a means to identify sets of specific TTPs leveraged within a Campaign (or in some way related to a Campaign).

- **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 Campaign data model references the Threat Actor data model as necessary to identify the Threat Actors who are potentially responsible for the Campaign (for the purpose of attribution) or who are in some way related to the Campaign. A

reference of the Threat Actor data model may also be used in a Campaign to capture the suspected intended effect of the Threat Actor.

3 STIX Campaign Data Model

The primary class of the STIX Campaign package is the `CampaignType` class, which characterizes a cyber threat campaign by capturing an asserted relationship between the threat actors who are involved, the TTPs used, and the incidents that comprise parts of the campaign in addition to other intrinsic properties. Similar to the primary classes of all of the component data models in STIX, the `CampaignType` class extends a base class defined in the STIX Common data model; more specifically, it extends the `CampaignBaseType` base class, which provides the essential identifier (`id`) and identifier reference (`idref`) properties.

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

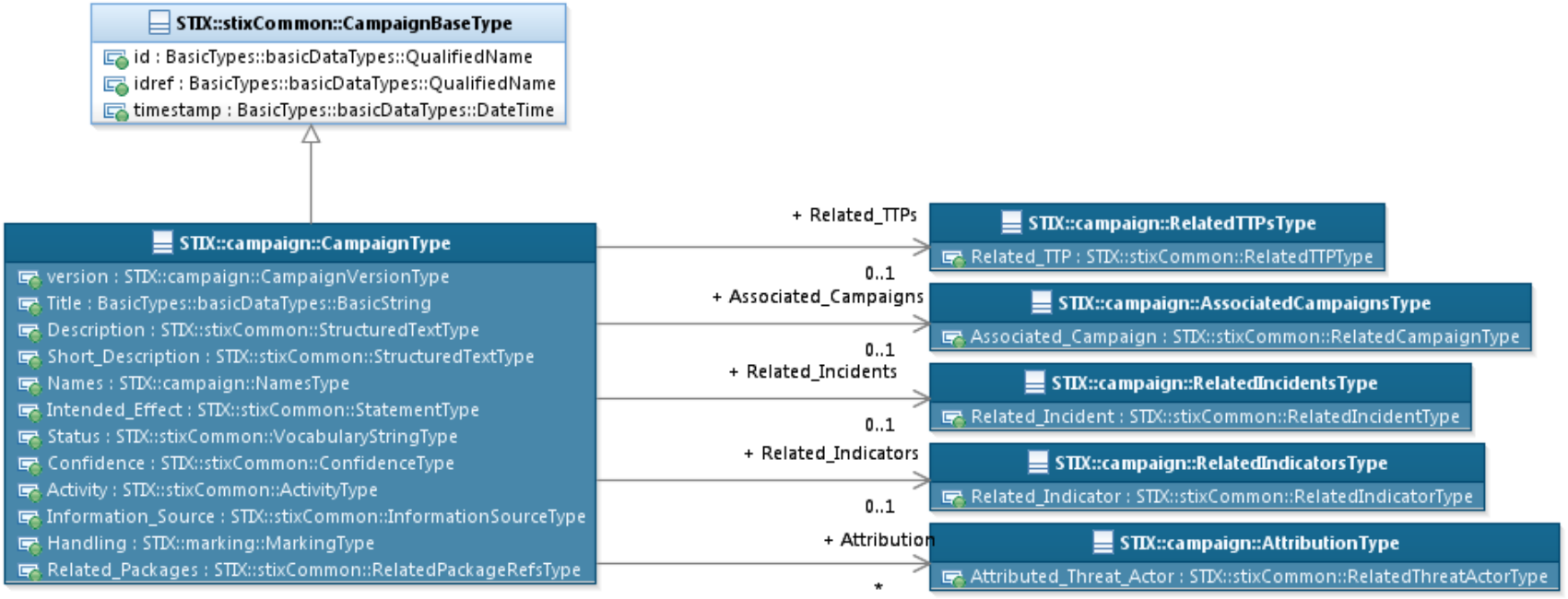

Figure 3-1. UML diagram of the `CampaignType` class

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

All classes defined in the Campaign data model are described in detail in Sections 3.1 through 3.7. Details are not provided for classes defined in non-Campaign data models; instead, the reader is refered 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 `CampaignType` class

Name	Type	Multiplicity	Description
version	`CampaignVersionType`	0..1	The `version` property specifies the version number of the STIX Campaign data model used to capture the information associated with the Campaign.
Title	`basicDataTypes:BasicString`	0..1	The `Title` property captures a title for the Campaign and reflects what the content producer thinks the Campaign as a whole should be called. The `Title` property is typically used by humans to reference a particular Campaign; however, it is not suggested for correlation.
Description	`stixCommon:StructuredTextType`	0..1	The `Description` property captures a textual description of the Campaign. Any length is permitted. Optional formatting is supported via the `structuring_format` property of the `StructuredTextType` class.
Short_Description	`stixCommon:StructuredTextType`	0..1	The `Short_Description` property captures a short textual description of the Campaign. This property is secondary and should only be used if the `Description` property is already populated and another, shorter description is available.
Names	`NamesType`	0..1	The `Names` property specifies a set of one or more names (i.e., aliases) used to identify the Campaign. An organization may use names that are created internally or externally (outside the organization). Note that the purpose of the

			Names property is different than that of the Title property: while the Title property is used to title the Campaign construct instance, the Names property gives the names of the set of activity that the Campaign describes.
Intended_Effect	stixCommon:StatementType	0..*	The Intended_Effect property characterizes the suspected effect that the Campaign is intended to have on its target(s), which includes a Value property that specifies the type of the effect. Examples of potential types include *theft*, *disruption*, and *unauthorized access* (these specific values are only provided to help explain the Value 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 types 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 Value property is '*IntendedEffectVocab-1.0*' (which is different than the default vocabulary provided for the StatementType class).
Status	stixCommon: VocabularyStringType	0..1	The Status property specifies the status of the Campaign. Examples of potential statuses include *ongoing*, *historical,* and *future* (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

			property is '*CampaignStatusVocab-1.0*'.
Related_TTPs	`RelatedTTPsType`	0..1	The `Related_TTPs` property specifies a set of one or more TTPs leveraged within the Campaign (or in some way related to a Campaign).
Related_Incidents	`RelatedIncidentsType`	0..1	The `RelatedIncidents` property specifies a set of one or more Incidents that are part of the Campaign (or in some way related to the Campaign).
Related_Indicators	`RelatedIndicatorsType`	0..1	The `Related_Indicators` property specifies a set of one or more Indicators relevant to the Campaign. *Note: as discussed in Section 3.5, this property is deprecated and is planned for removal in STIX Campaign Version 2.0.*
Attribution	`AttributionType`	0..*	The `Attribution` property specifies attribution information in the form of a set of one or more Threat Actors who are asserted to be responsible for the Campaign. *Multiple groups can be captured by defining multiple Attribution elements*.
Associated_Campaigns	`AssociatedCampaignsType`	0..1	The `Associated_Campaigns` property specifies a set of one or more other Campaigns related to this Campaign.
Confidence	`stixCommon:ConfidenceType`	0..1	The `Confidence` property characterizes the level of confidence in the accuracy of the collection of information captured for the Campaign.
Activity	`stixCommon:ActivityType`	0..*	The `Activity` property characterizes a defender activity associated with the Campaign. Its underlying abstract class must be extended to include the chosen format of activity characterization.
Information_Source	`stixCommon: InformationSourceType`	0..1	The `Information_Source` property characterizes the source of the Campaign information. Examples of details captured include identitifying characteristics, time-related attributes, and a list of the tools used to collect the information.

Handling	`marking:MarkingType`	0..1	The `Handling` property specifies data handling markings for the properties of this Campaign. The marking scope is limited to the campaign and the content it contains. Note that data handling markings can also be specified at a higher level.
Related_Packages	`stixCommon: RelatedPackagesRefsType`	0..1	The `Related_Packages` property specifies a set of one or more STIX Packages that are related to the Campaign.

3.1 CampaignVersionType Enumeration

The `CampaignVersionType` enumeration is an inventory of all versions of the Campaign data model, all of which are valid in STIX Version 1.1.1.

The enumeration literals are given in Table 3-2.

Table 3-2. Values of the `CampaignVersionType` enumeration

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

3.2 NamesType Class

The `NamesType` class specifies a set of one or more names used to identify the Campaign. Note that an equivalent `NamesType` class is defined in the STIX Common data model; this duplication is due to backward-compatiblity issues and will be corrected in the

next major release of STIX[3]. At that time, the `campaign:NamesType` class will be removed, and Campaign names will be defined via the `stixCommon:NamesType` class.

The property of the `NamesType` class is shown in Table 3-3.

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

Name	Type	Multiplicity	Description
Name	`stixCommon: VocabularyStringType`	1..*	The `Name` property is used to specify a single name or alias that identifies the Campaign. 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. No default vocabulary class has been defined for STIX 1.1.1.

3.3 RelatedTTPsType Class

The `RelatedTTPsType` class specifies a set of one or more TTPs asserted to be leveraged within the Campaign (or in some way related to a Campaign). 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 `RelatedTTPsType` class is shown in Figure 3-2.

[3] Essentially, the first version of the `NamesType` class was defined within the Campaign data model to allow users to capture the names by which a campaign is known. However, when the relationship between a Campaign and an Indicator was moved from the Campaign data model to the Indicator data model, users still needed the ability to refer to a Campaign by name. Existing policy of not having one component data model (Indicator) depend on another (Campaign) meant that an equivalent `NamesType` class was added to the STIX Common data model. In the next major version of STIX, it is expected that the `NamesType` class will be removed from the Campaign data model and that all Campaign names will be defined via the STIX Common `NamesType` class.

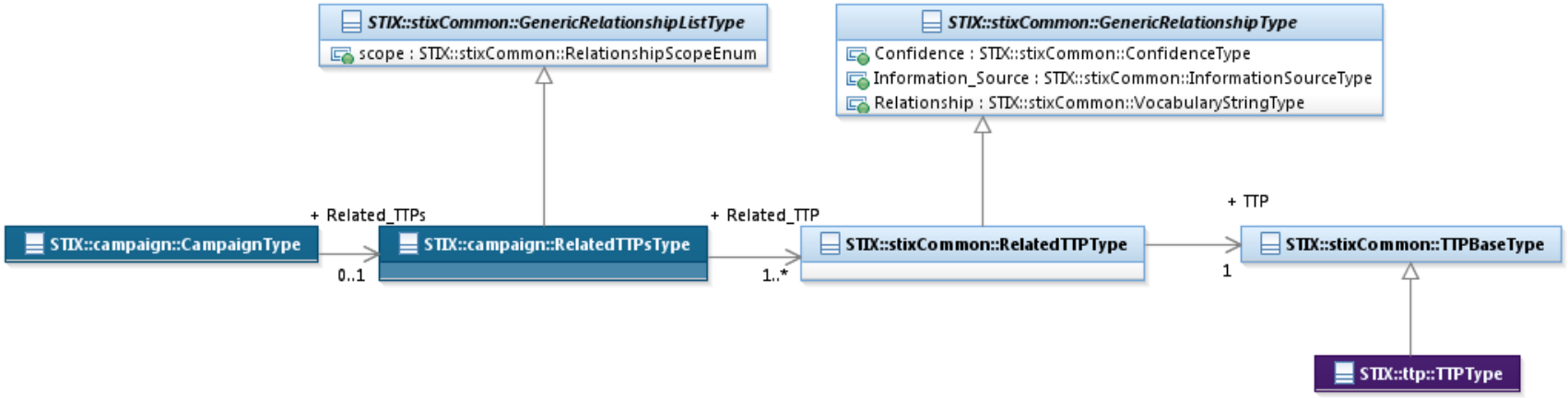

Figure 3-2. UML diagram of the `RelatedTTPsType` class

Table 3-4 shows the properties of the `RelatedTTPsType` specialization and is associated with the UML diagram given in Figure 3-2.

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

Name	Type	Multiplicity	Description
Related_TTP	`stixCommon:RelatedTTPType`	1..*	The `Related_TTP` property specifies a TTP leveraged within the Campaign (or in some way related to a Campaign) and characterizes the relationship between the Campaign and the TTP by capturing information such as the level of confidence that the Campaign and the TTP are related, the source of the relationship information, and the type of relationship.

3.4 RelatedIncidentsType Class

The `RelatedIncidentsType` class specifies a set of one or more Incidents asserted as part of the Campaign (or in some way related to the Campaign). 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 `RelatedIncidentsType` class is shown in Figure 3-3.

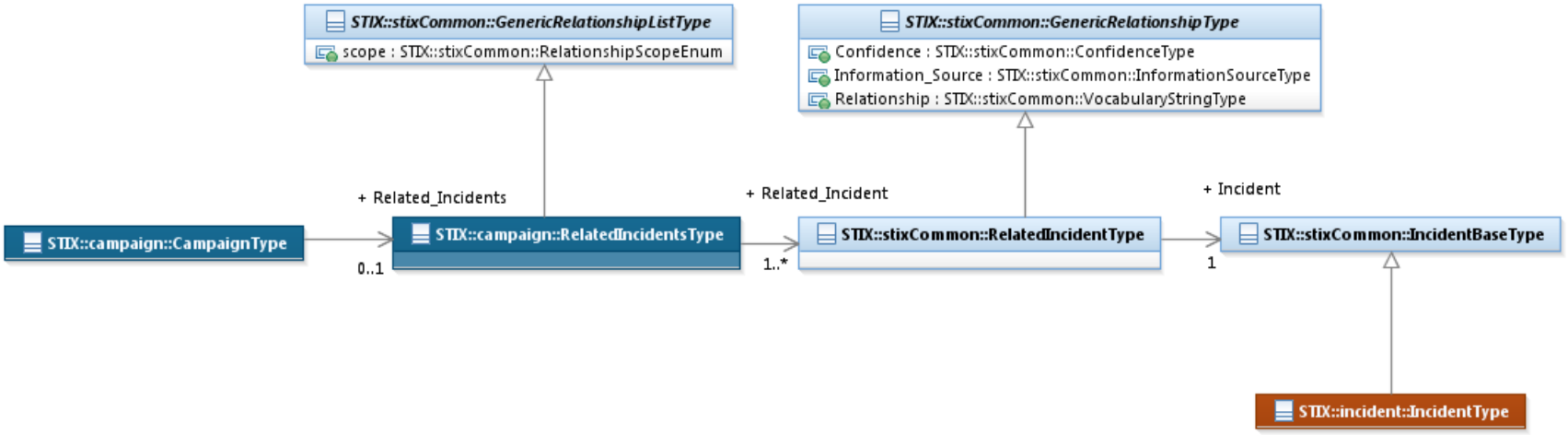

Figure 3-3. UML diagram of the `RelatedIncidentsType` class

Table 3-5 shows the properties of the `RelatedIncidentType` specialization and is associated with the UML diagram of Figure 3-3.

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

Name	Type	Multiplicity	Description
Related_Incident	`stixCommon:RelatedIncidentType`	1..*	The `Related_Incident` property specifies an Incident asserted as part of the Campaign (or in some way related to the Campaign) and characterizes the relationship between the Campaign and the Incident by capturing information such as the level of confidence that the Campaign and the Incident are related, the source of the relationship information, and the type of relationship.

3.5 RelatedIndicatorsType Class (deprecated)

The `RelatedIndicatorsType` class specifies a set of one or more Indicators asserted as relevent to a Campaign. 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 3-4.

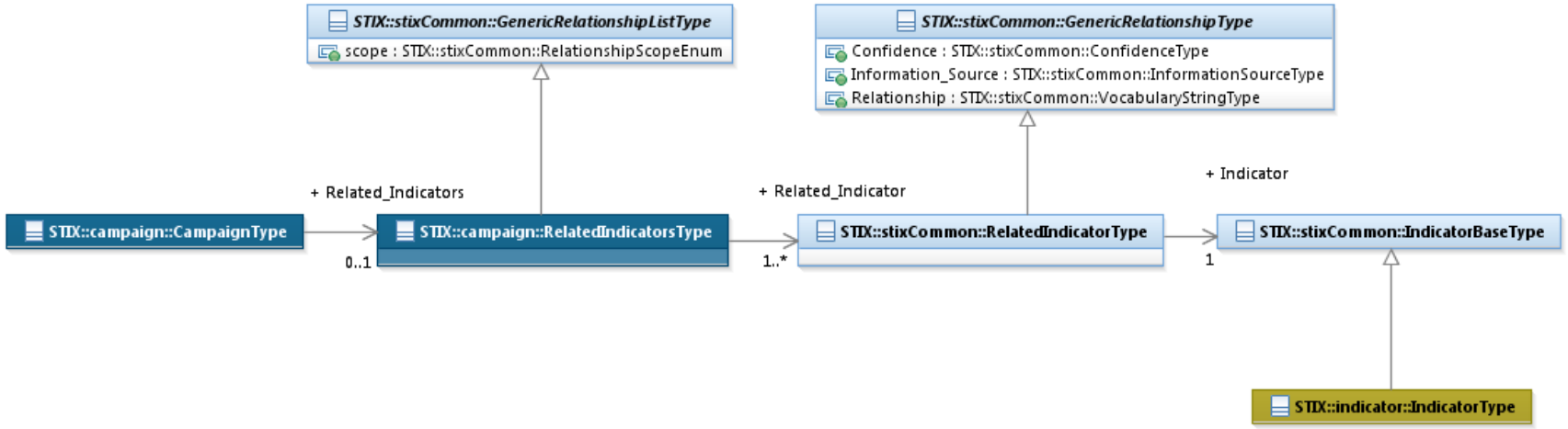

Figure 3-4. UML diagram of the `RelatedIndicatorsType` class

***NOTE**: The `Related_Indicators` property of the `CampaignType` was deprecated in STIX Version 1.1, and it is slated for removal in STIX Version 2.0 (it remains in Version 1.1.1 of the Campaign data model for backward compatibility). Therefore, because no other property requires it, the `RelatedIndicatorsType` class will be removed in Version 2.0 of the Campaign data model. Unless legacy code or content require the use of the `Related_Indicators` property, Relationships between Indicators and Campaigns in STIX v1.1.1 SHOULD be represented using the `Related_Campaigns` property of the `indicator:IndicatorType` class.*

Table 3-6 (shaded to indicate deprecation) shows the properties of the `RelatedIncidentType` specialization and is associated with the UML diagram of Figure 3-4.

Table 3-6. Properties of the `RelatedIndicatorsType` class (deprecated)

Name	Type	Multiplicity	Description
Related_Indicator	`stixCommon:RelatedIncidentType`	1..*	The `Related_Indicator` property specifies an Indicator asserted as relevant to this Campaign and

			characterizes the relationship between the Indicator and the Campaign by capturing information such as the level of confidence that the Indicator and the Campaign are related, the source of the relationship information, and the type of relationship.

3.6 AttributionType Class

The `AttributionType` class specifies a set of one or more Threat Actors asserted as related to a Campaign from an attribution perspective. 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 `AttributionType` class is shown in Figure 3-5.

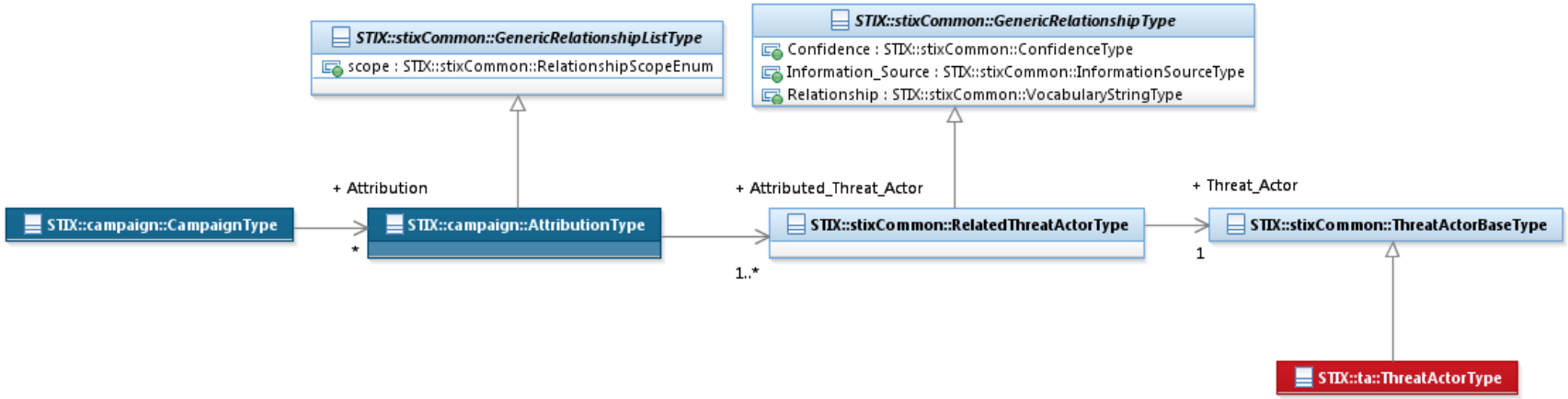

Figure 3-5. UML diagram of the `AttributionType` class

Table 3-7 shows the properties of the `AttributionType` specialization and is associated with the UML diagram in Figure 3-5.

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

Name	Type	Multiplicity	Description
Attributed_Threat_Actor	`stixCommon:RelatedThreatActorType`	1..*	The `Attributed_Threat_Actor` property specifies a Threat Actor asserted as related to the Campaign and characterizes the relationship between the Threat Actor and the Campaign by capturing information such as the level of confidence that the Threat Actor and the Campaign are related, the source of the relationship information, and the type of the relationship.

3.7 AssociatedCampaignsType Class

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

The UML diagram corresponding to the `AssociatedCampaignsType` class is shown in Figure 3-6.

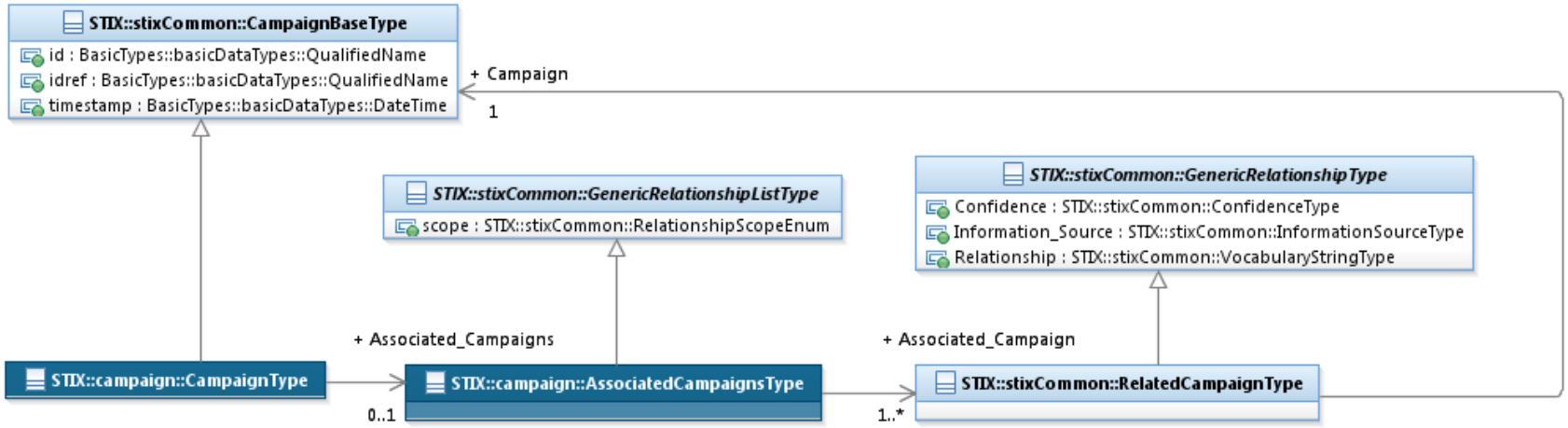

Figure 3-6. UML Diagram of the `AssociatedCampaignsType` class

Table 3-8 shows the properties of the `AssociatedCampaignType` specialization and is associated with the UML diagram in Figure 3-6.

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

Name	Type	Multiplicity	Description
Associated_Campaign	`stixCommon:RelatedCampaignType`	1..*	The `Associated_Campaign` property specifies another Campaign associated with this Campaign and characterizes the relationship between the Campaigns by capturing information such as the level of confidence that the Campaigns are related, the source of the relationship information, and type of the relationship. A relationship between Campaigns may represent assertions of general associativity or different versions of the same Campaign.

Appendix – XML Implementation

The initial implementation for STIX v1.1.1 uses XML schema as a structured mechanism for detailed discussion, collaboration and refinement among the communities involved. The complete listing of XML representation resources can be found on the STIX website [REL].

References

References made in this document are listed below.

[REL] STIX™ Campaign Model as implement in XSD
https://stix.mitre.org/language/version4.1/xxx_schema.xsd

[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

[$STIX_O$] STIX™ 1.1.1 Specification Overview
http://stix.mitre.org/about/documents/XXXX.pdf

[$STIX_{INC}$] STIX™ 1.1.1 Incident 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

[$STIX_{TA}$] STIX™ 1.1.1 Threat Actor Specification (v1.1.1)
http://stix.mitre.org/about/documents/XXXX.pdf

[$STIX_{TTP}$] STIX™ 1.1.1 TTP Specification (v1.1.1)
http://stix.mitre.org/about/documents/XXXX.pdf

[TOU] Terms of Use
http://stix.mitre.org/about/termsofuse.html