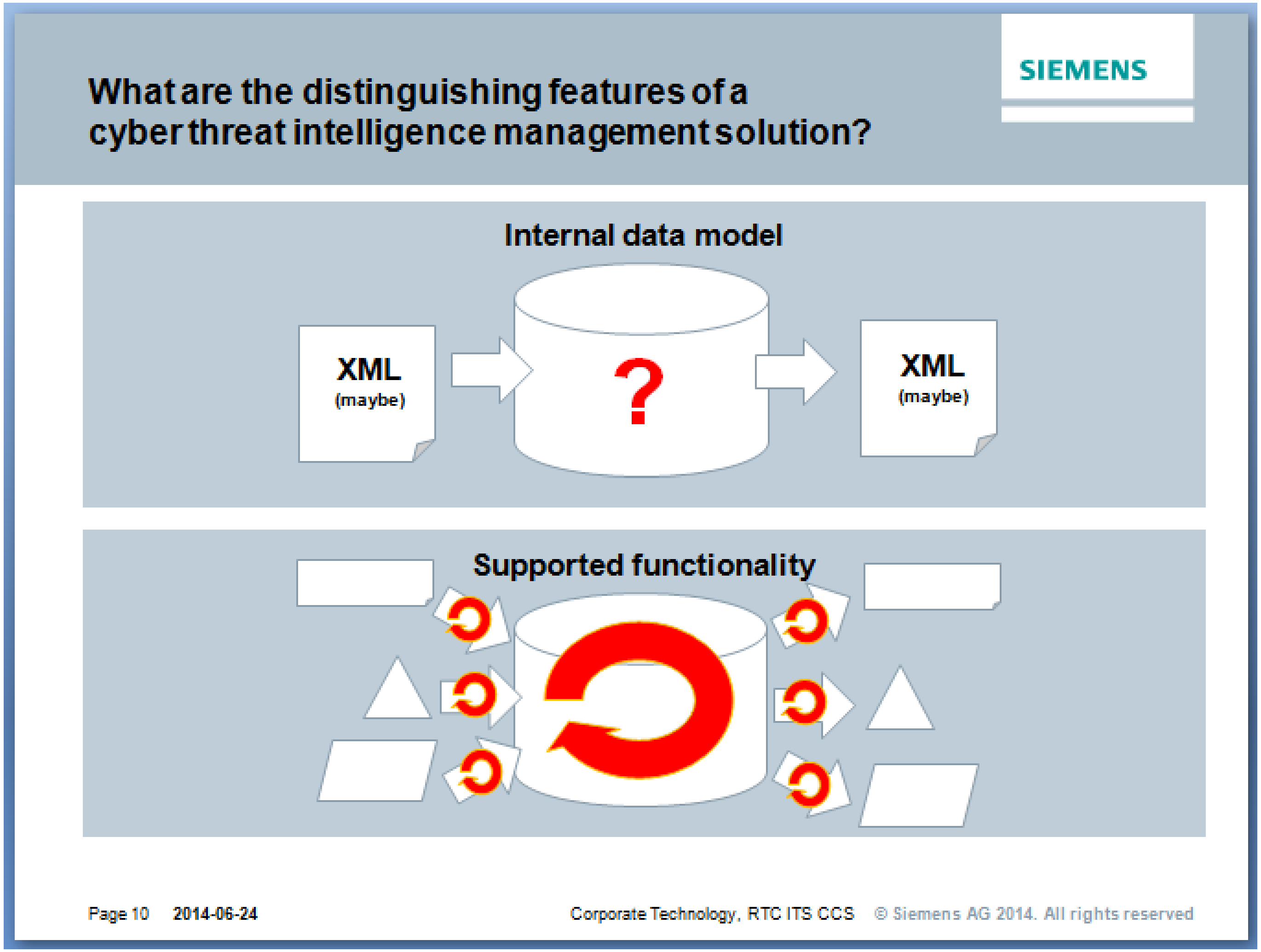
SIEMENS

FIRST TC on Threat Intelligence 2016 | Dr. Bernd Grobauer

Theory and Practice of TI Mgmt. Using STIX and Cybox: Musings on the Data Model

siemens.com/answers

A slide from FIRST 2014 (Boston)



What are the distinguishing features of a cyber threat intelligence management solution?

Internal data model

XML
(maybe)

?

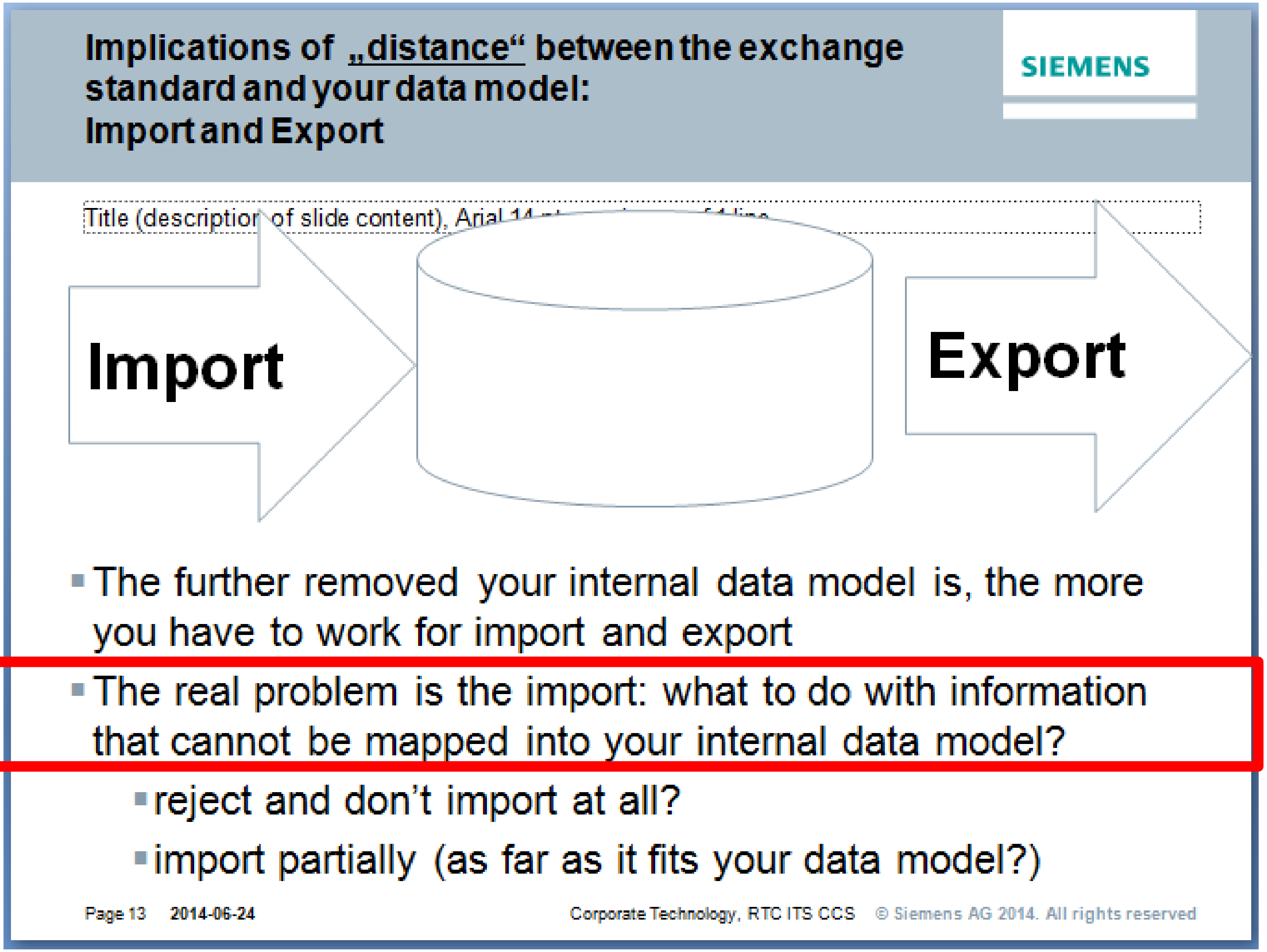
XML
(maybe)

Supported functionality

Page 10 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

SIEMENS

A slide from FIRST 2014 (Boston)



SIEMENS

Implications of „distance“ between the exchange standard and your data model: Import and Export

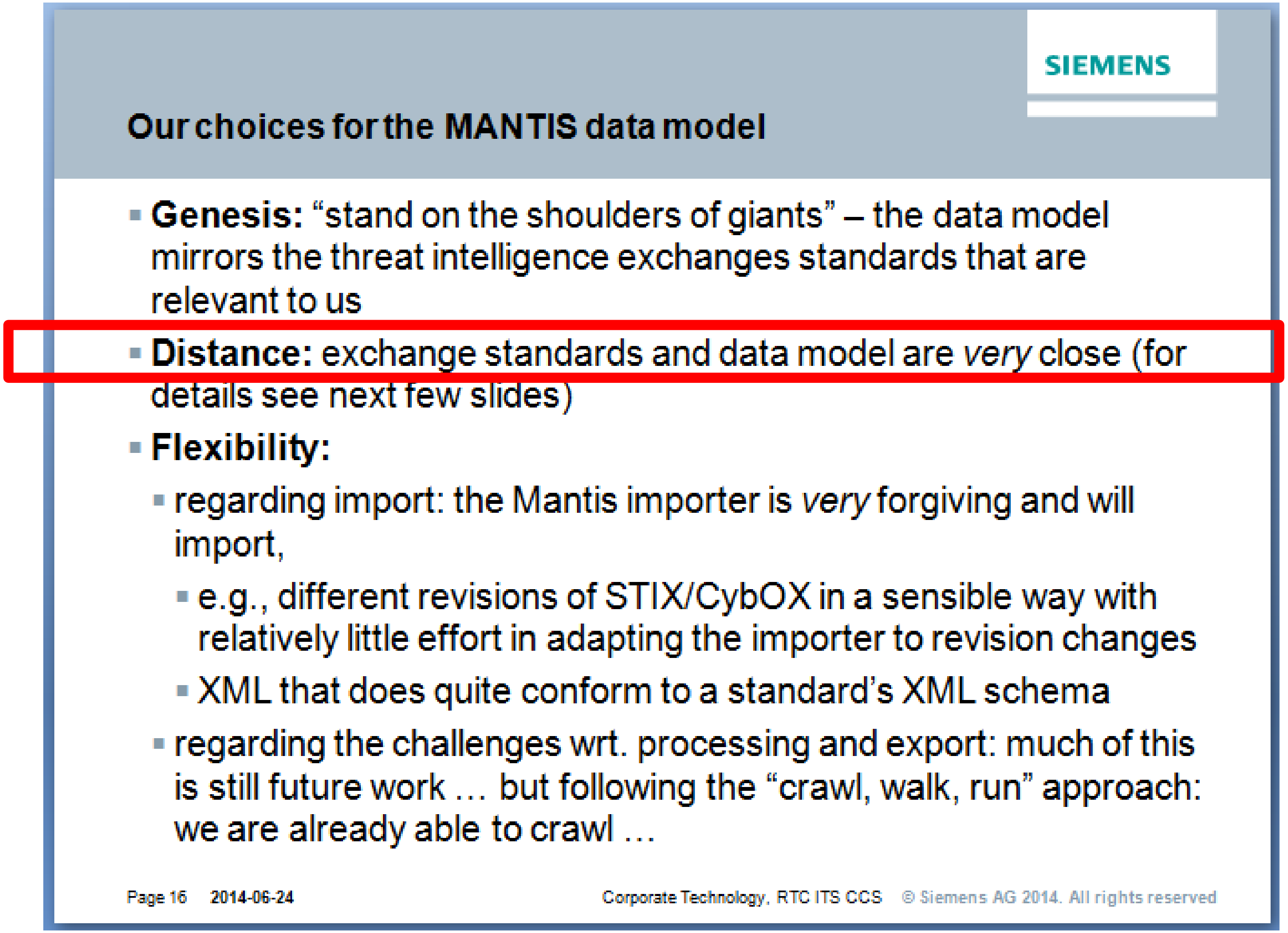
Title (description of slide content), Arial 14 pt

Import
Export

- The further removed your internal data model is, the more you have to work for import and export
- The real problem is the import: what to do with information that cannot be mapped into your internal data model?
  - reject and don't import at all?
  - import partially (as far as it fits your data model?)

Page 13 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

A slide from FIRST 2014 (Boston)



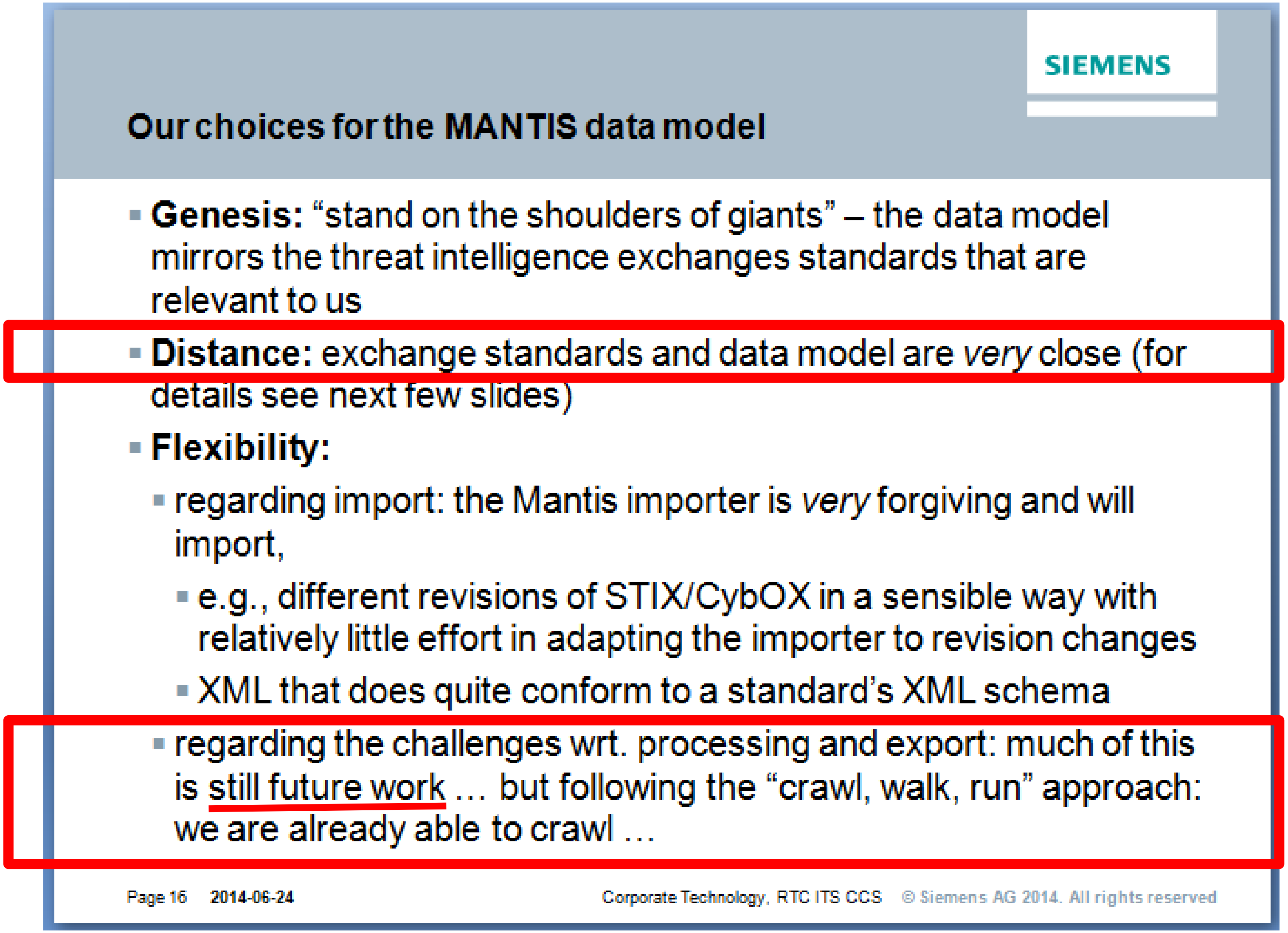
SIEMENS

Our choices for the MANTIS data model

- **Genesis:** “stand on the shoulders of giants” – the data model mirrors the threat intelligence exchanges standards that are relevant to us
- **Distance:** exchange standards and data model are *very* close (for details see next few slides)
- **Flexibility:**
  - regarding import: the Mantis importer is *very* forgiving and will import,
    - e.g., different revisions of STIX/CybOX in a sensible way with relatively little effort in adapting the importer to revision changes
    - XML that does quite conform to a standard’s XML schema
  - regarding the challenges wrt. processing and export: much of this is still future work … but following the “crawl, walk, run” approach: we are already able to crawl …

Page 16 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

A slide from FIRST 2014 (Boston)

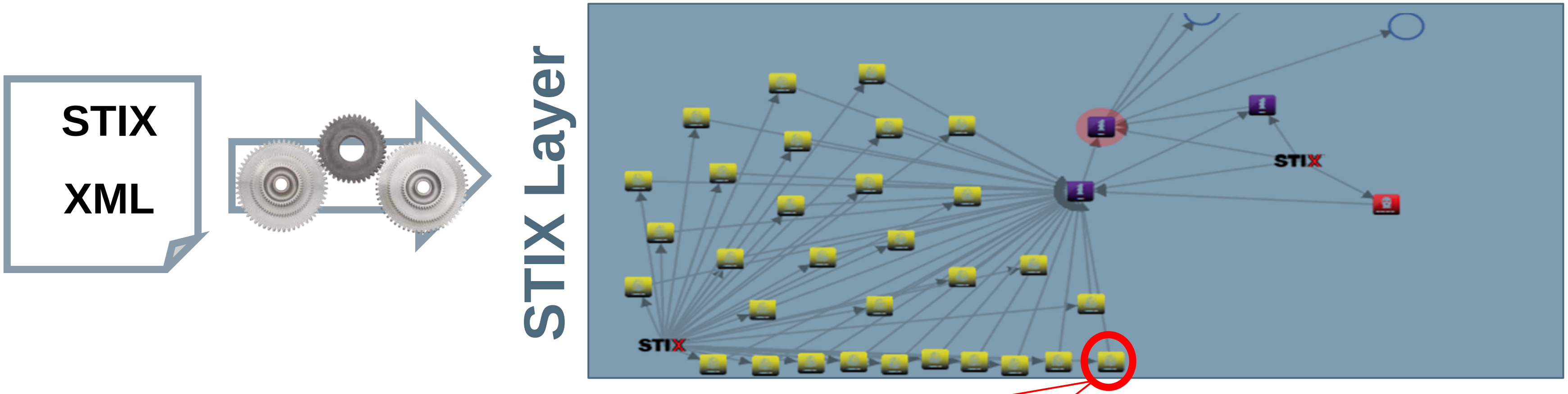


Our choices for the MANTIS data model

- **Genesis:** “stand on the shoulders of giants” – the data model mirrors the threat intelligence exchanges standards that are relevant to us
- **Distance:** exchange standards and data model are *very* close (for details see next few slides)
- **Flexibility:**
  - regarding import: the Mantis importer is *very* forgiving and will import,
    - e.g., different revisions of STIX/CybOX in a sensible way with relatively little effort in adapting the importer to revision changes
    - XML that does quite conform to a standard’s XML schema
  - regarding the challenges wrt. processing and export: much of this is still future work … but following the “crawl, walk, run” approach: we are already able to crawl …

Page 16 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

MANTIS STIX Layer
Turn STIX/CybOX XML into interconnected „InfoObjects“



STIX

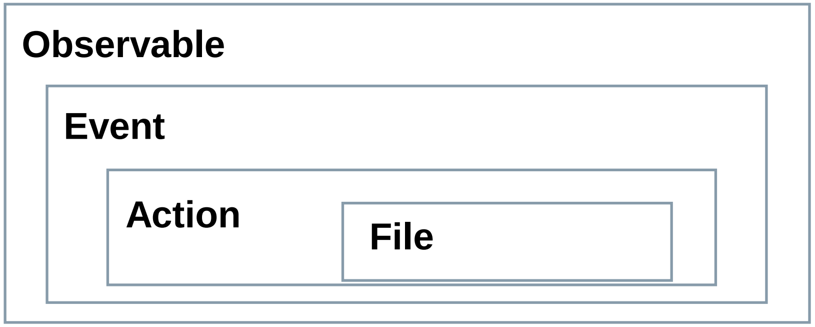
XML

STIX Layer

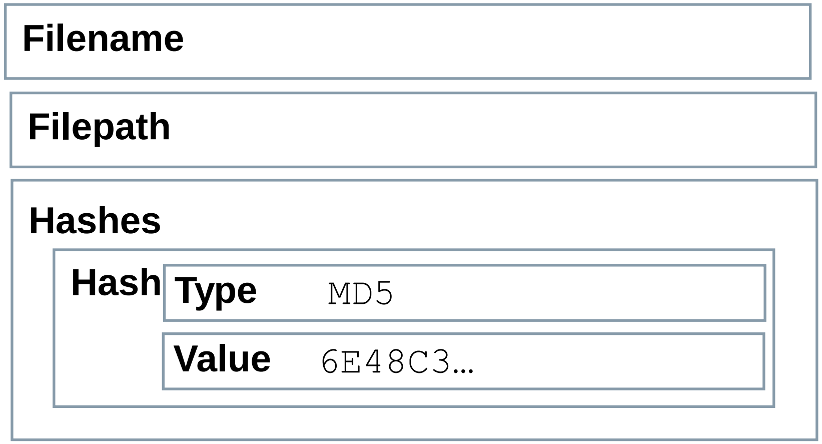
Each „InfoObject“ represented by list of „facts“ that closely reflect XML structure

Relationships and Facts in STIX/CybOX

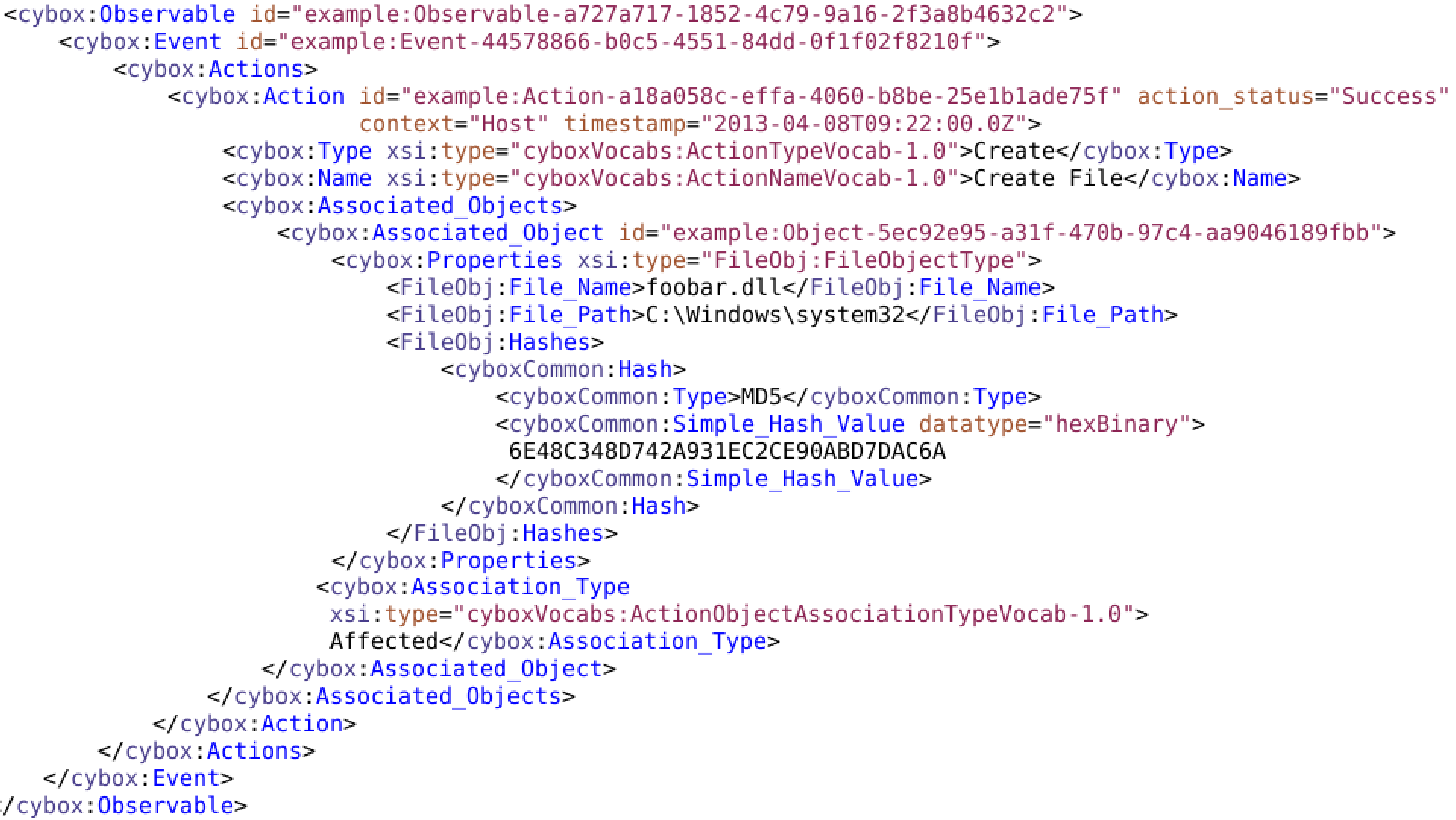
- If you look at STIX and CybOX, you see that XML’s hierarchical structure is used for two different purposes:
  - modeling of containment relations between different objects



  - structuring of facts

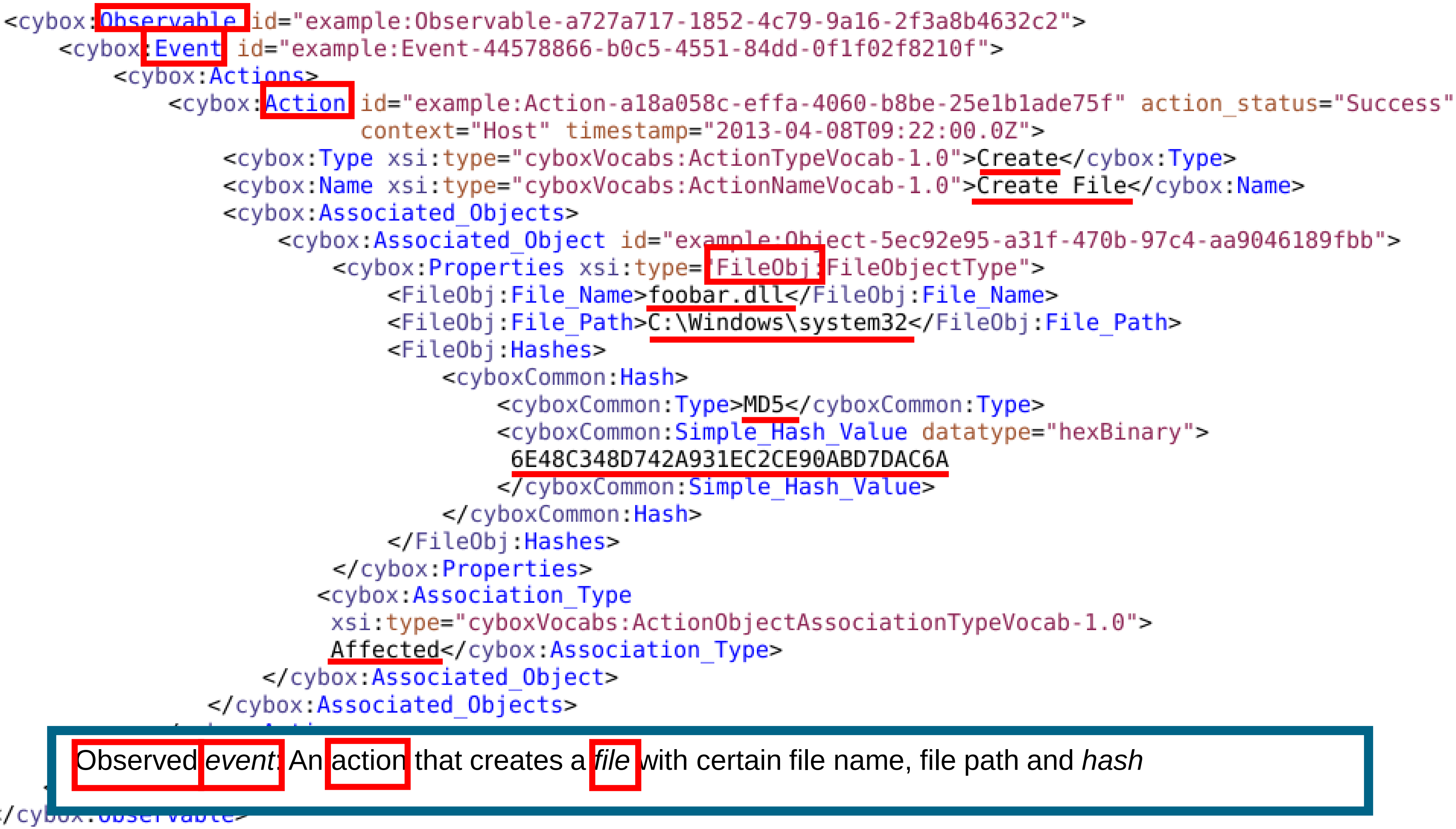


Example: A CybOX Observable XML Source



```
<cybox:Observable id="example:Observable-a727a717-1852-4c79-9a16-2f3a8b4632c2">
    <cybox:Event id="example:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">
        <cybox:Actions>
            <cybox:Action id="example:Action-a18a058c-effa-4060-b8be-25e1b1ade75f" action_status="Success"
                          context="Host" timestamp="2013-04-08T09:22:00.0Z">
                <cybox:Type xsi:type="cyboxVocabs:ActionTypeVocab-1.0">Create</cybox:Type>
                <cybox:Name xsi:type="cyboxVocabs:ActionNameVocab-1.0">Create File</cybox:Name>
                <cybox:Associated_Objects>
                    <cybox:Associated_Object id="example:Object-5ec92e95-a31f-470b-97c4-aa9046189fbb">
                        <cybox:Properties xsi:type="FileObj:FileObjectType">
                            <FileObj:File_Name>foobar.dll</FileObj:File_Name>
                            <FileObj:File_Path>C:\Windows\system32</FileObj:File_Path>
                            <FileObj:Hashes>
                                <cyboxCommon:Hash>
                                    <cyboxCommon:Type>MD5</cyboxCommon:Type>
                                    <cyboxCommon:Simple_Hash_Value datatype="hexBinary">
                                     6E48C348D742A931EC2CE90ABD7DAC6A
                                    </cyboxCommon:Simple_Hash_Value>
                                </cyboxCommon:Hash>
                            </FileObj:Hashes>
                        </cybox:Properties>
                       <cybox:Association_Type
                        xsi:type="cyboxVocabs:ActionObjectAssociationTypeVocab-1.0">
                        Affected</cybox:Association_Type>
                   </cybox:Associated_Object>
               </cybox:Associated_Objects>
           </cybox:Action>
       </cybox:Actions>
   </cybox:Event>
</cybox:Observable>
```

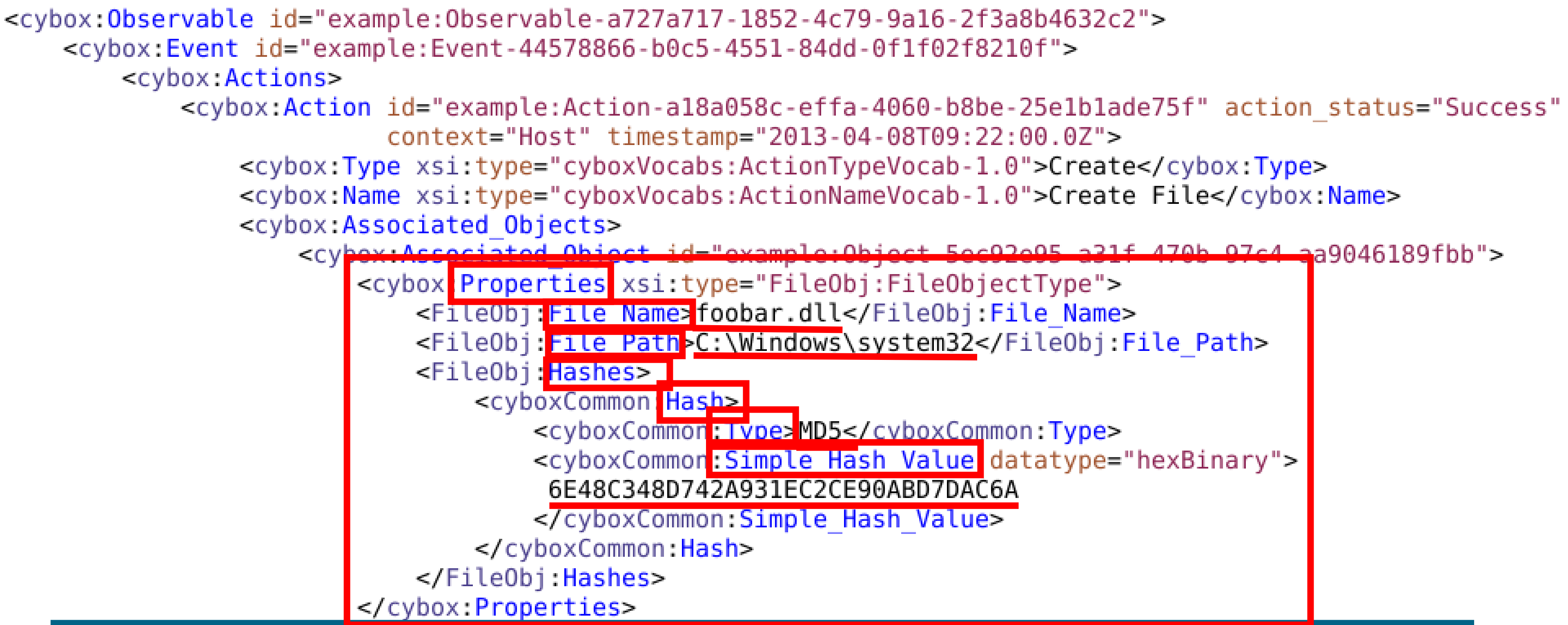
Example: Importing a CybOX 2.0 Observable XML Source: Focusing on objects and facts



```
<cybox:Observable id="example:Observable-a727a717-1852-4c79-9a16-2f3a8b4632c2">
    <cybox:Event id="example:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">
        <cybox:Actions>
            <cybox:Action id="example:Action-a18a058c-effa-4060-b8be-25e1b1ade75f" action_status="Success"
                          context="Host" timestamp="2013-04-08T09:22:00.0Z">
                <cybox:Type xsi:type="cyboxVocabs:ActionTypeVocab-1.0">Create</cybox:Type>
                <cybox:Name xsi:type="cyboxVocabs:ActionNameVocab-1.0">Create File</cybox:Name>
                <cybox:Associated_Objects>
                    <cybox:Associated_Object id="example:Object-5ec92e95-a31f-470b-97c4-aa9046189fbb">
                        <cybox:Properties xsi:type="FileObj:FileObjectType">
                            <FileObj:File_Name>foobar.dll</FileObj:File_Name>
                            <FileObj:File_Path>C:\Windows\system32</FileObj:File_Path>
                            <FileObj:Hashes>
                                <cyboxCommon:Hash>
                                    <cyboxCommon:Type>MD5</cyboxCommon:Type>
                                    <cyboxCommon:Simple_Hash_Value datatype="hexBinary">
                                     6E48C348D742A931EC2CE90ABD7DAC6A
                                    </cyboxCommon:Simple_Hash_Value>
                                </cyboxCommon:Hash>
                            </FileObj:Hashes>
                        </cybox:Properties>
                       <cybox:Association_Type
                        xsi:type="cyboxVocabs:ActionObjectAssociationTypeVocab-1.0">
                        Affected</cybox:Association_Type>
                   </cybox:Associated_Object>
               </cybox:Associated_Objects>
            </cybox:Action>
/cybox:Observable>
```

Observed *event*: An action that creates a *file* with certain file name, file path and *hash*

Example: A CybOX Observable XML Source
Extracting „flat“ facts from hierarchical XML



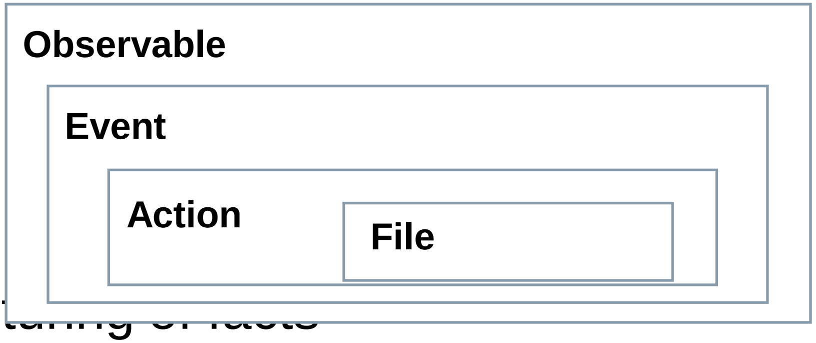
```
<cybox:Observable id="example:Observable-a727a717-1852-4c79-9a16-2f3a8b4632c2">
    <cybox:Event id="example:Event-44578866-b0c5-4551-84dd-0f1f02f8210f">
        <cybox:Actions>
            <cybox:Action id="example:Action-a18a058c-effa-4060-b8be-25e1b1ade75f" action_status="Success"
                          context="Host" timestamp="2013-04-08T09:22:00.0Z">
                <cybox:Type xsi:type="cyboxVocabs:ActionTypeVocab-1.0">Create</cybox:Type>
                <cybox:Name xsi:type="cyboxVocabs:ActionNameVocab-1.0">Create File</cybox:Name>
                <cybox:Associated_Objects>
                    <cybox:Associated_Object id="example:Object-5ec92e95-a31f-470b-97c4-aa9046189fbb">
                        <cybox:Properties xsi:type="FileObj:FileObjectType">
                            <FileObj:File_Name>foobar.dll</FileObj:File_Name>
                            <FileObj:File_Path>C:\Windows\system32</FileObj:File_Path>
                            <FileObj:Hashes>
                                <cyboxCommon:Hash>
                                    <cyboxCommon:Type>MD5</cyboxCommon:Type>
                                    <cyboxCommon:Simple_Hash_Value datatype="hexBinary">
                                     6E48C348D742A931EC2CE90ABD7DAC6A
                                    </cyboxCommon:Simple_Hash_Value>
                                </cyboxCommon:Hash>
                            </FileObj:Hashes>
                        </cybox:Properties>
/cy
```

The facts we are really interested into about the observed file are:

- Properties/File_Name = foobar.dll
- Properties/File_Path = C:\Windows\system32
- *Properties/Hashes/Hash/Type = MD5*
- *Properties/Hashes/Hash/Simple_Hash_Value = 6E48C34D74A931EC2CE90ABD7DAC6A*

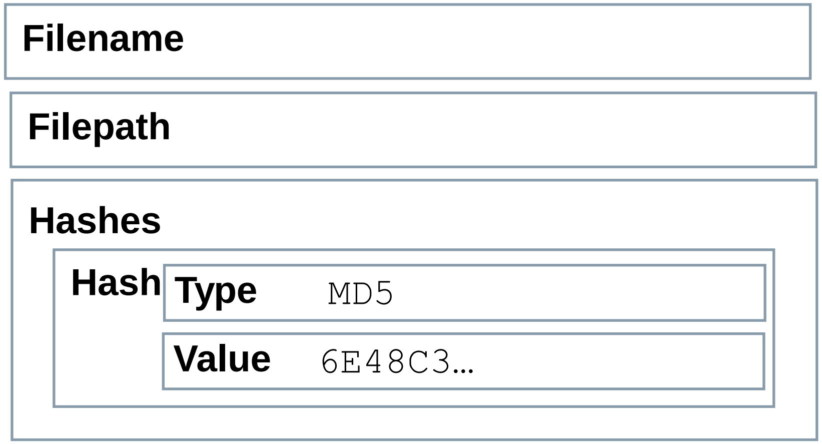
Relationships and Facts in STIX/CybOX

- If you look at STIX and CybOX, you see that XML's hierarchical structure is used for two different purposes:
  - modeling of containment relations between different objects



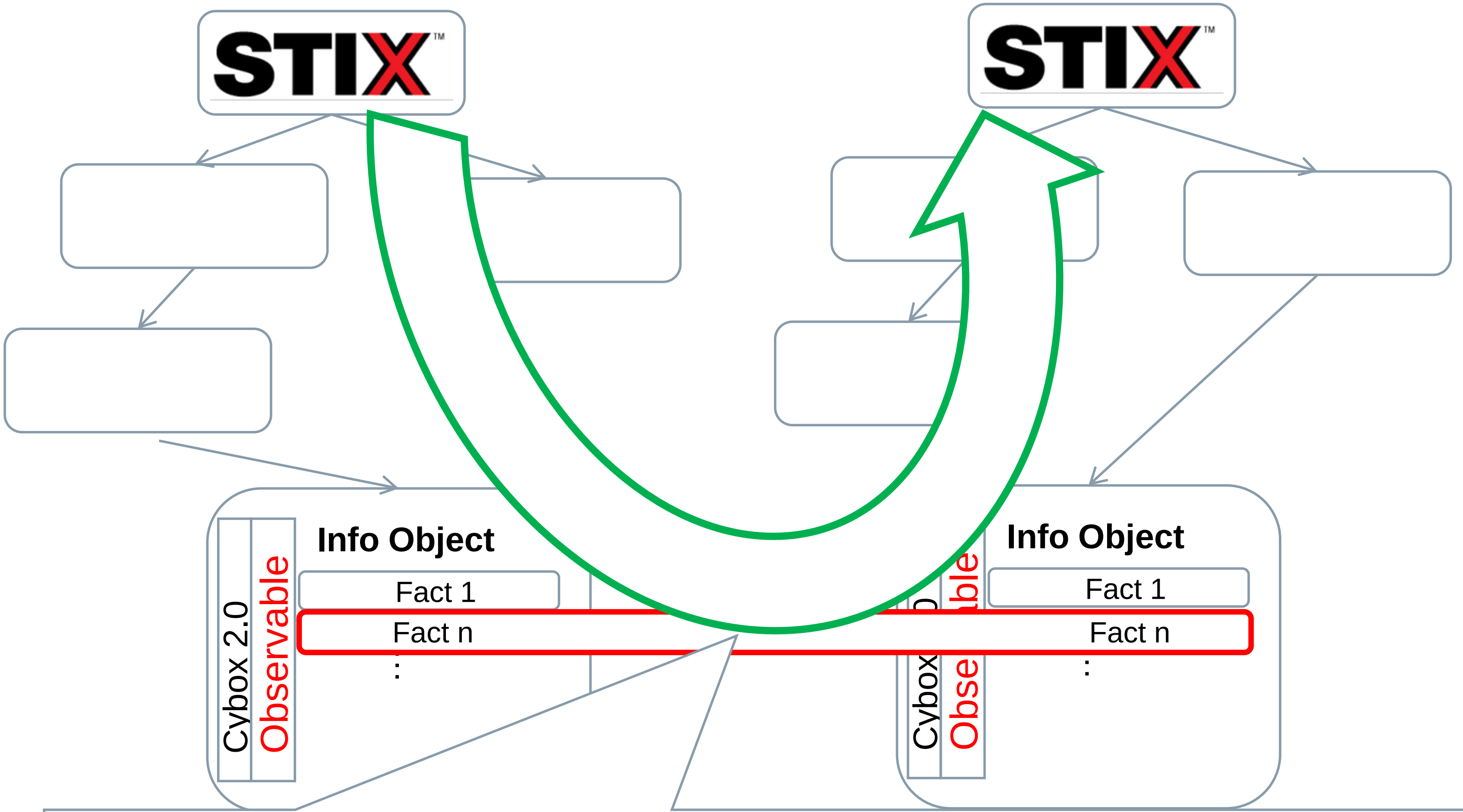
This leads to nodes and edges

  - structuring of facts



This leads to facts about a node

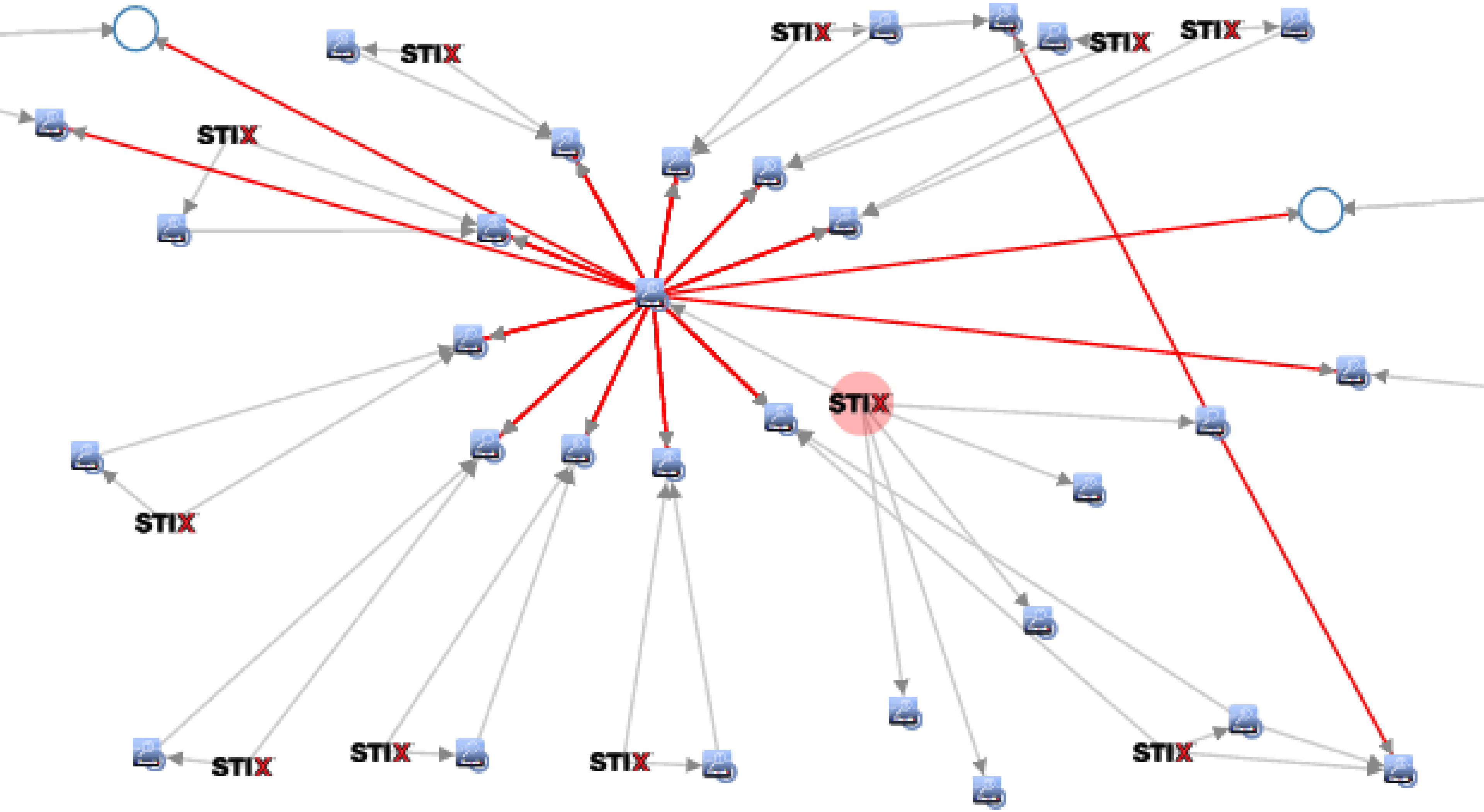
Correlation by Facts using the MANTIS data model



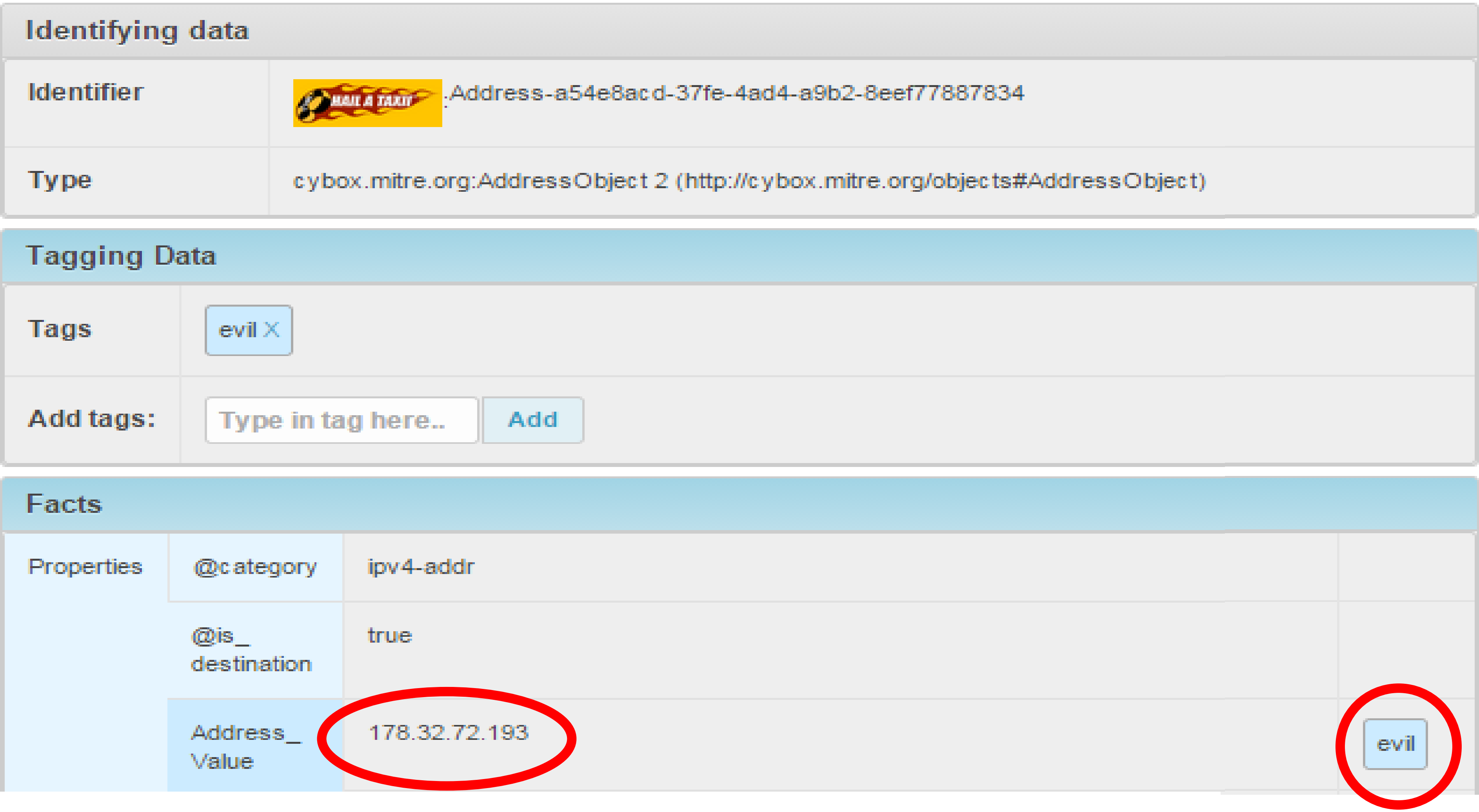
Properties/Hashes/Hash/SimpleHashValue=6E48C3… is shared between two different InfoObjects

SIEMENS

STIX-Layer feature of MANTIS: Correlation on facts



STIX-Layer feature of MANTIS: „Fact-based“ Tagging



Identifying data

Identifier	.Address-a54e8acd-37fe-4ad4-a9b2-8eef77887834
Type	cybox.mitre.org:AddressObject 2 (http://cybox.mitre.org/objects#AddressObject)

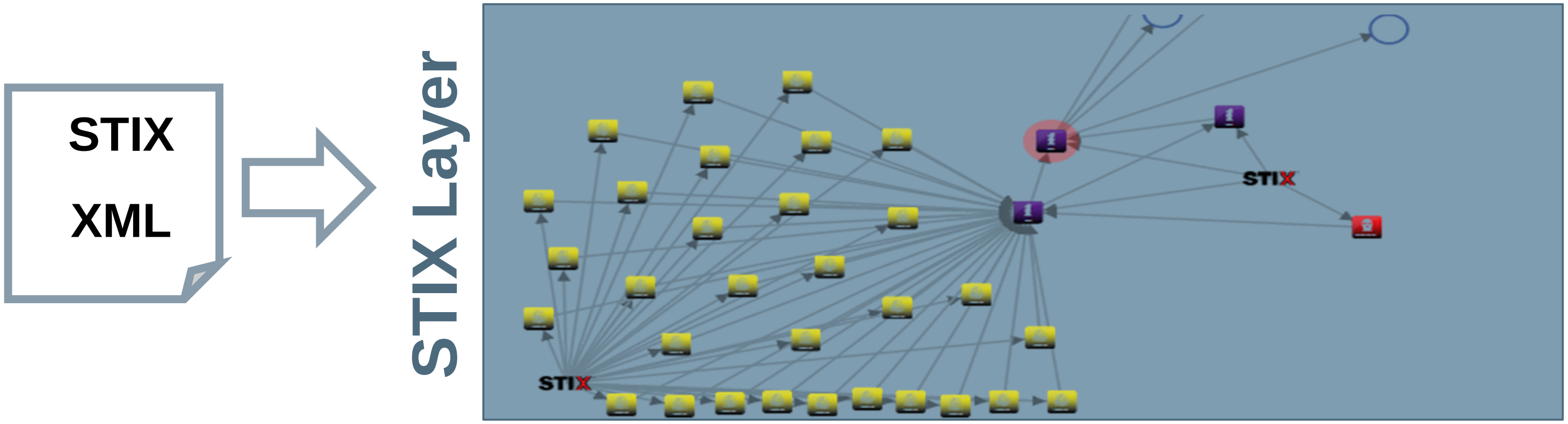
Tagging Data

Tags	evil ✕
Add tags:	Type in tag here.. Add

Facts

Properties	@category	ipv4-addr	
	@is_ destination	true	
	Address_ Value	178.32.72.193	evil

The GUI Problem



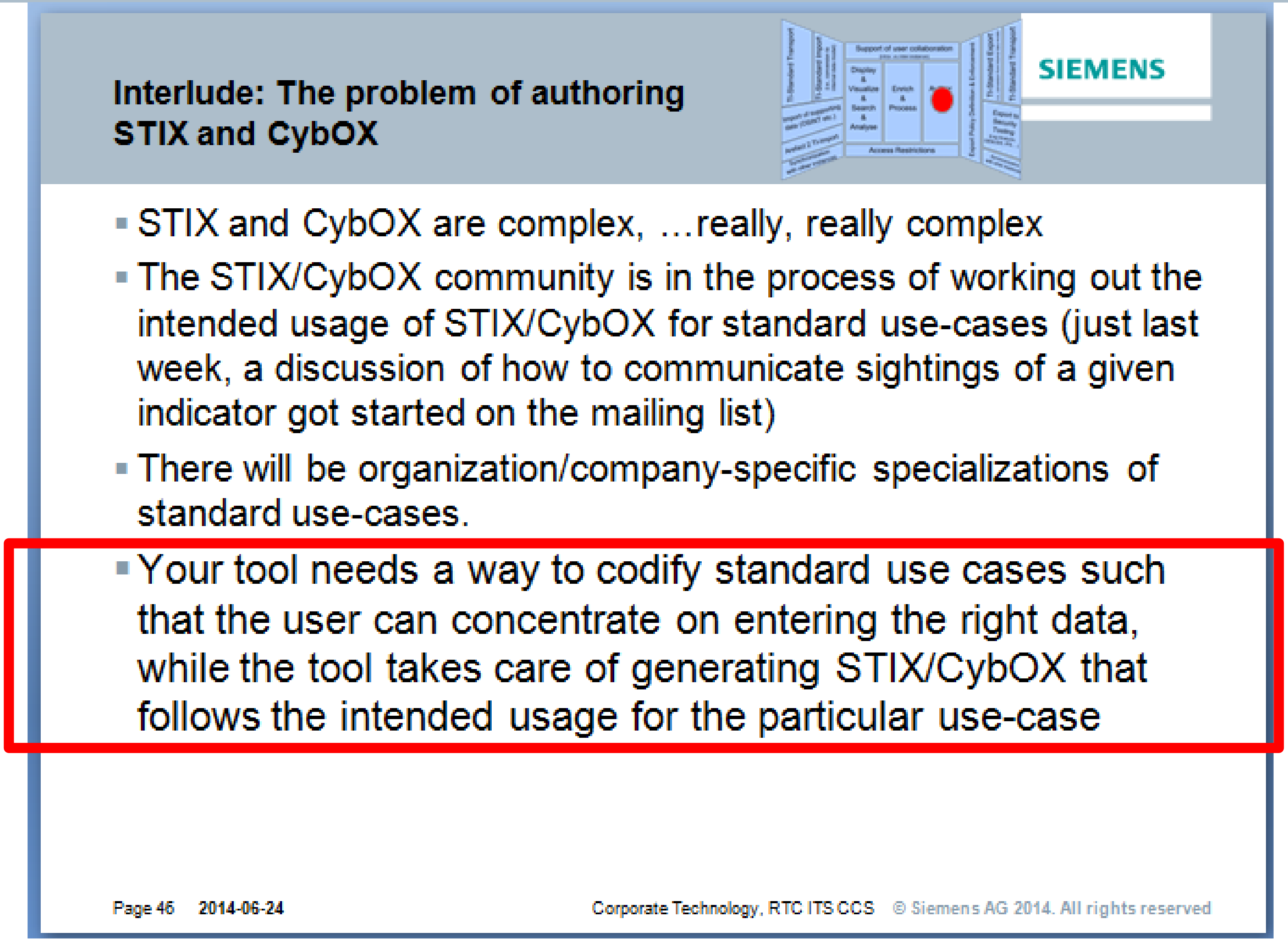
STIX

XML

STIX Layer

SIEMENS

The GUI Problem …


Support of user collaboration
Display & Visualize & Search & Analyze
Enrich & Process
Access Restrictions

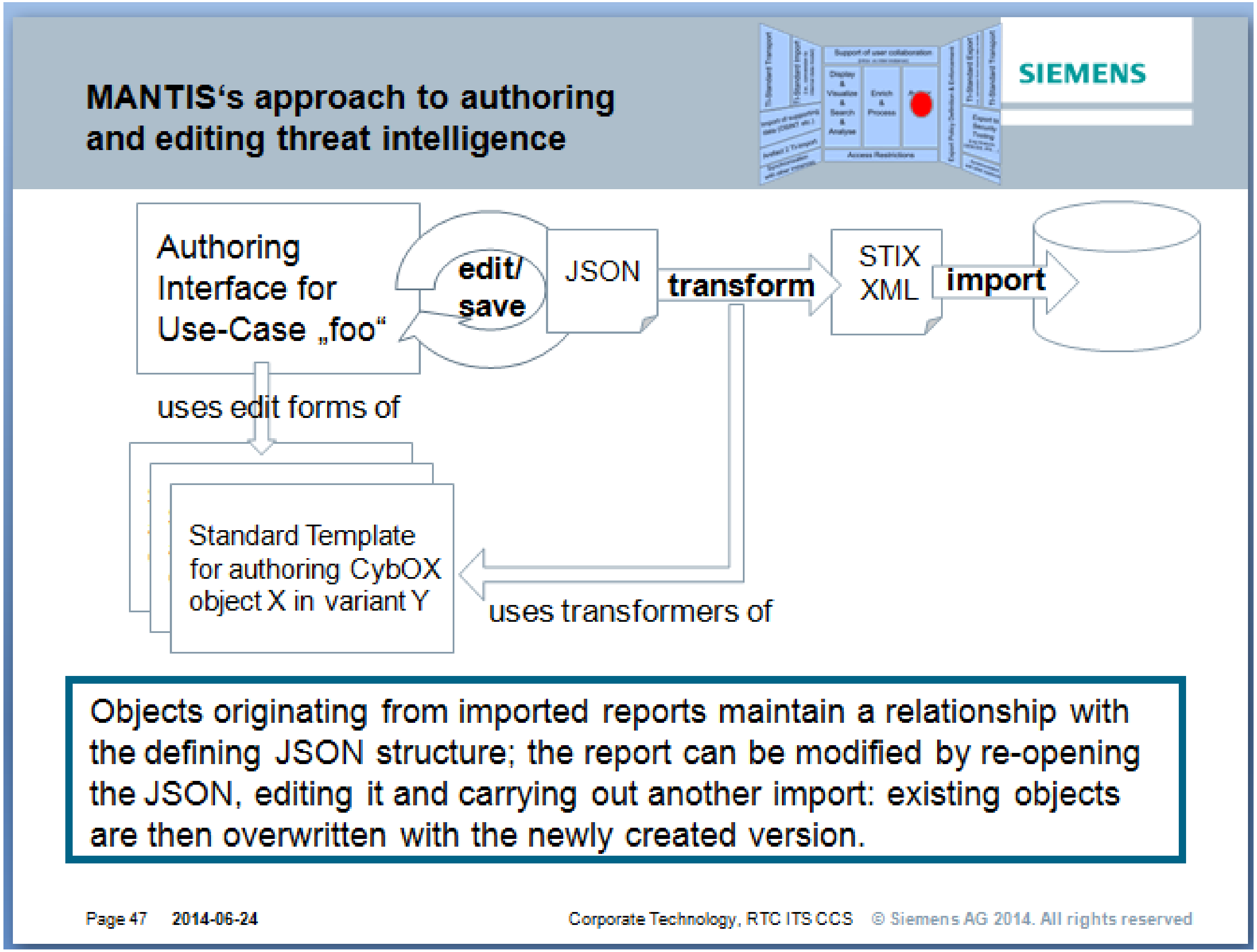
SIEMENS

Interlude: The problem of authoring STIX and CybOX

- STIX and CybOX are complex, …really, really complex
- The STIX/CybOX community is in the process of working out the intended usage of STIX/CybOX for standard use-cases (just last week, a discussion of how to communicate sightings of a given indicator got started on the mailing list)
- There will be organization/company-specific specializations of standard use-cases.
- Your tool needs a way to codify standard use cases such that the user can concentrate on entering the right data, while the tool takes care of generating STIX/CybOX that follows the intended usage for the particular use-case

Page 46 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

MANTIS‘ GUI approach



MANTIS‘s approach to authoring and editing threat intelligence

Authoring Interface for Use-Case „foo“

edit/ save

JSON

transform

STIX XML

import

uses edit forms of

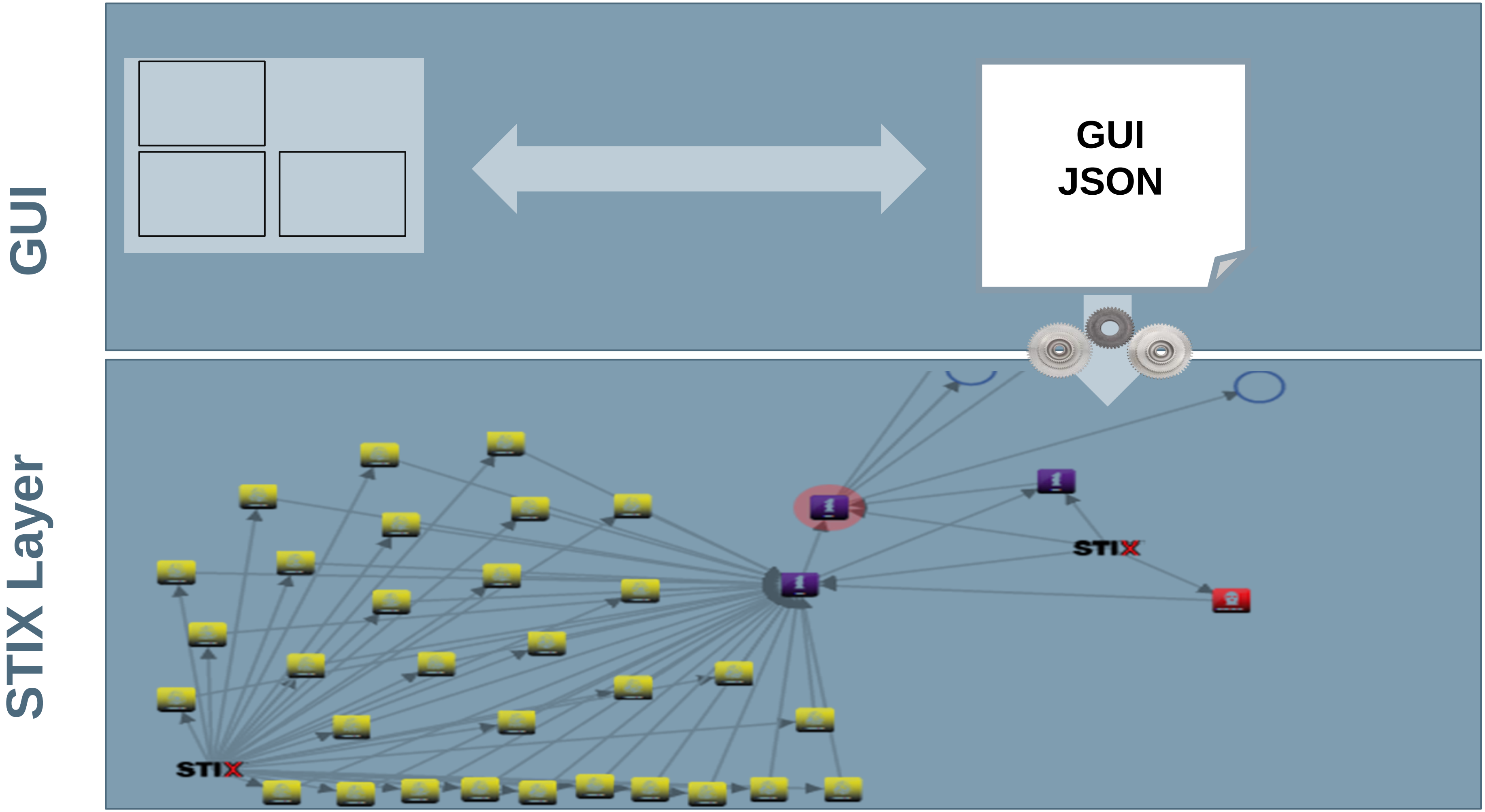
Standard Template for authoring CybOX object X in variant Y

uses transformers of

Objects originating from imported reports maintain a relationship with the defining JSON structure; the report can be modified by re-opening the JSON, editing it and carrying out another import: existing objects are then overwritten with the newly created version.

Page 47 2014-06-24 Corporate Technology, RTC ITS CCS © Siemens AG 2014. All rights reserved

The GUI Problem: Now we have two layers



SIEMENS

The GUI Problem: Now we have two layers

- **Analyst has two views on the same data**
  - Authoring view
  - STIX-based view

Consequences:

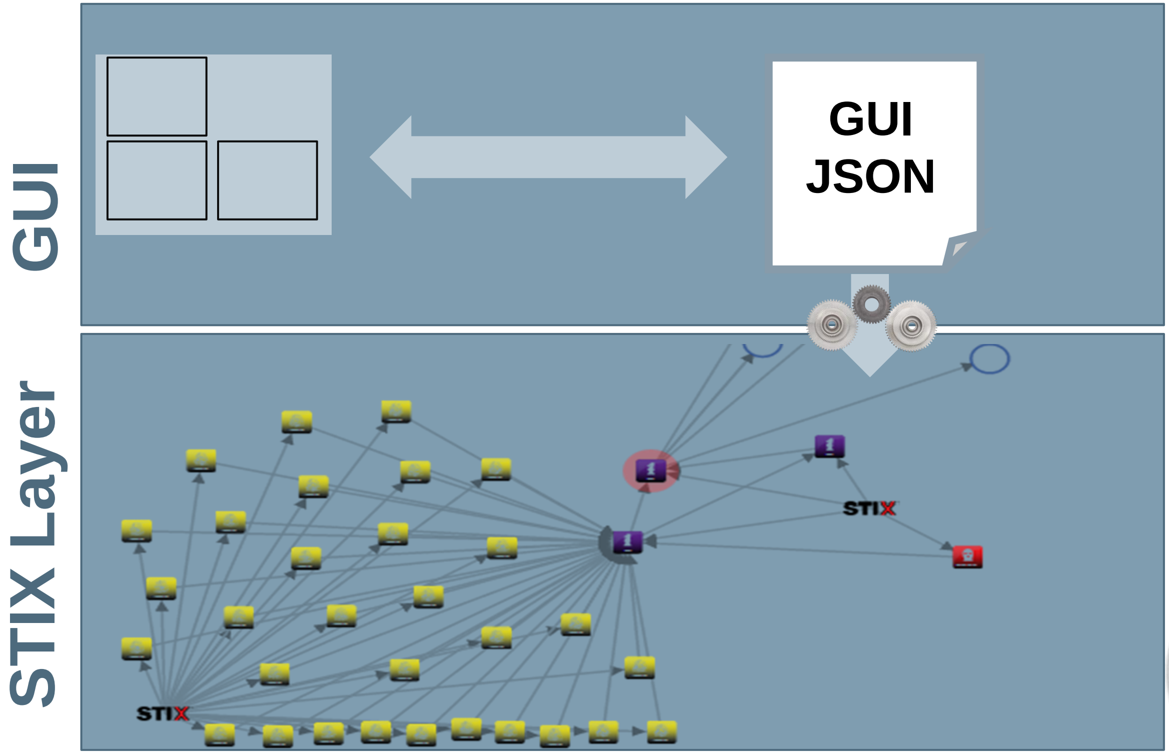
- This adds some complexity and makes lives of users/analysts harder than woudl be the case for directly working on the model
- Danger of divergencies between GUI layer and STIX layer if subsequent import after changes is not carried out

SIEMENS

The GUI Problem: Now we have two layers

- **Only „own“ GUI-made reports can be edited via GUI**
  - You CANNOT work on a report received as STIX XML, since there is no way back from STIX to „GUI JSON“
  - You could think about „per entity/object“ editing support (which allows you to edit the entity/object „features“ the GUI author chose to implement … but the more features you support, the more complicated things get again.

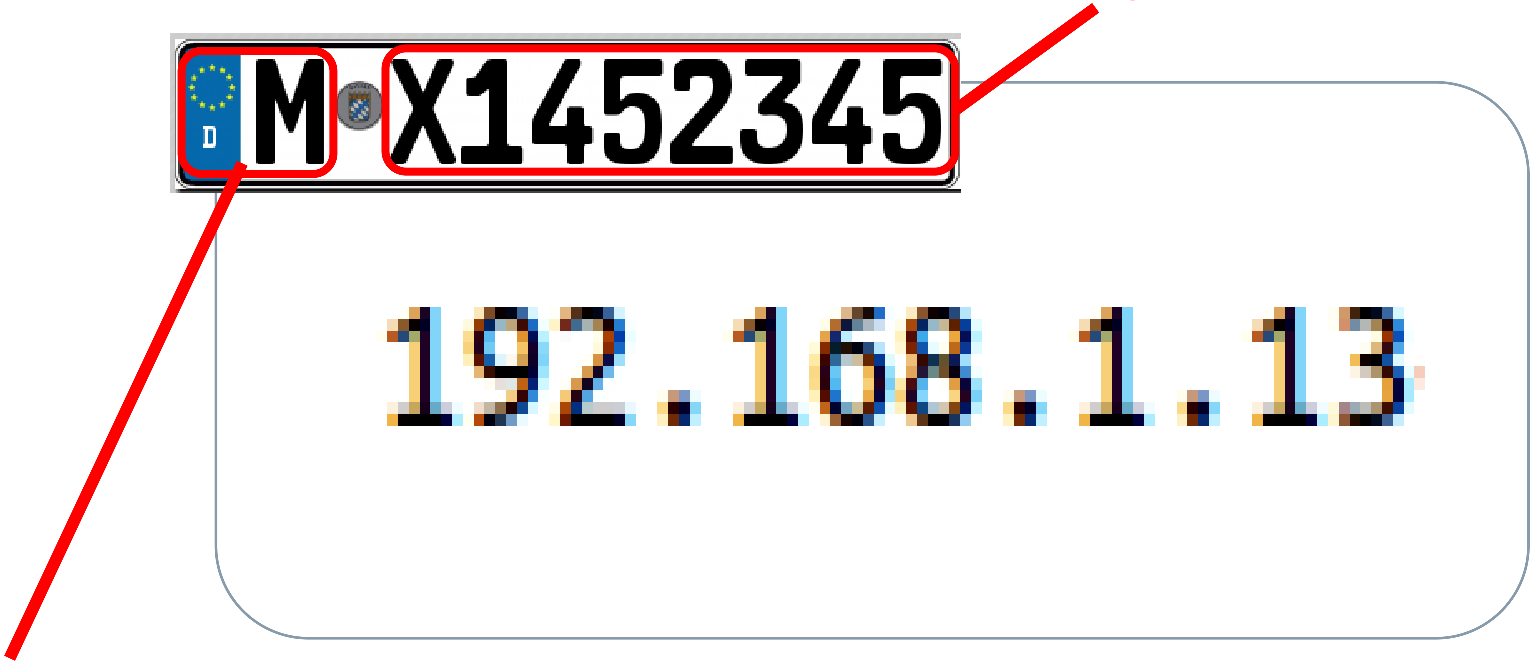
The „making stuff actionable“ problem



? What are the „top“ needles (IPs, URLs, Hashes, …) I want to look for in my haystack?

The STIX/CybOX Duplication/Multiplication Problem: Munich CERT has observed 192.168.1.13

UID



Namespace: munich.de

Others have observed 192.168.1.13 as well (Berlin CERT even four times in four different reports)

192.168.1.13

192.168.1.13

192.168.1.13

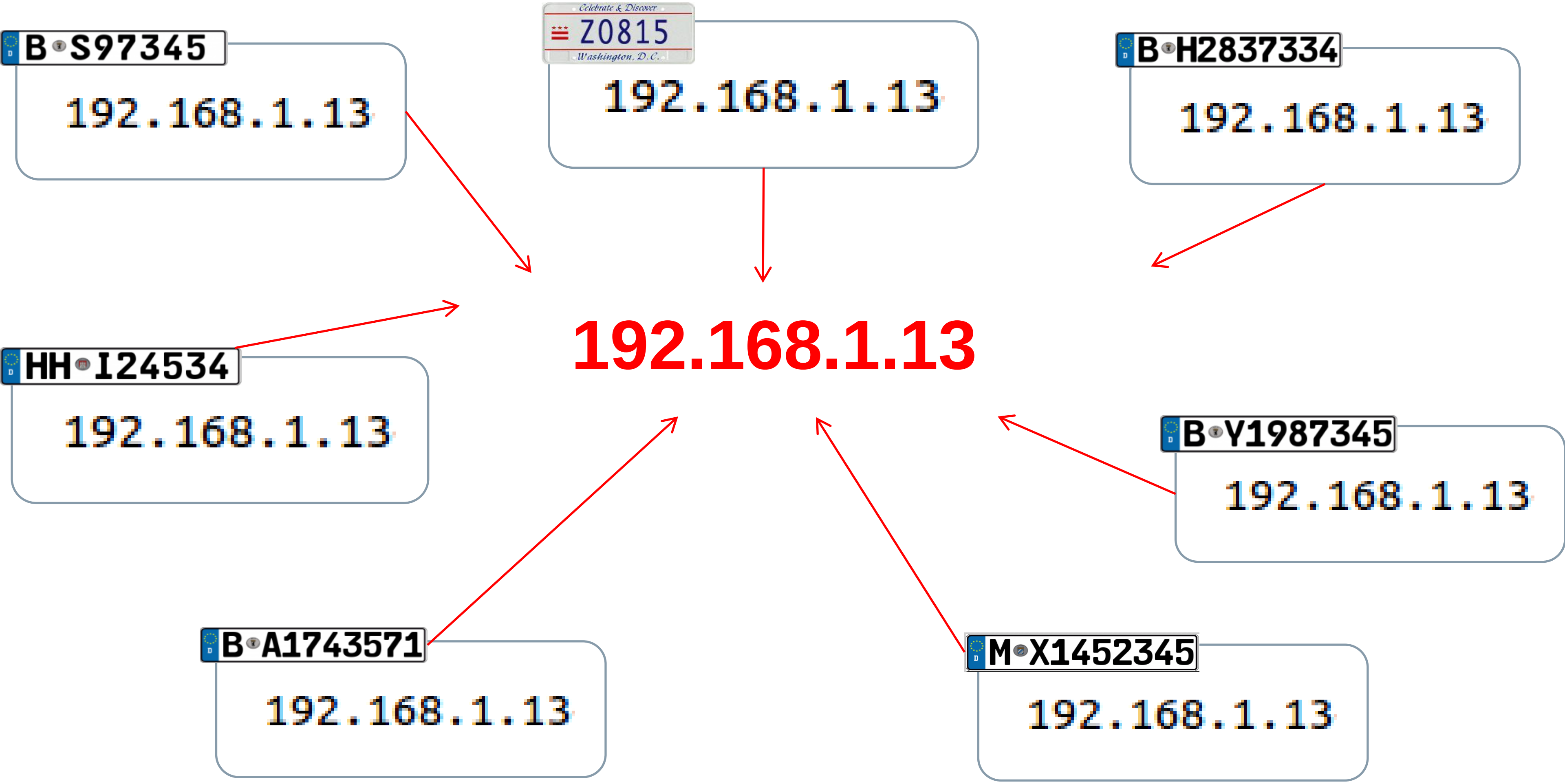
192.168.1.13

192.168.1.13

192.168.1.13

192.168.1.13

To get anywhere close to finding your top needles, you need some canoncial representation of your needles



192.168.1.13

192.168.1.13

192.168.1.13

192.168.1.13

192.168.1.13

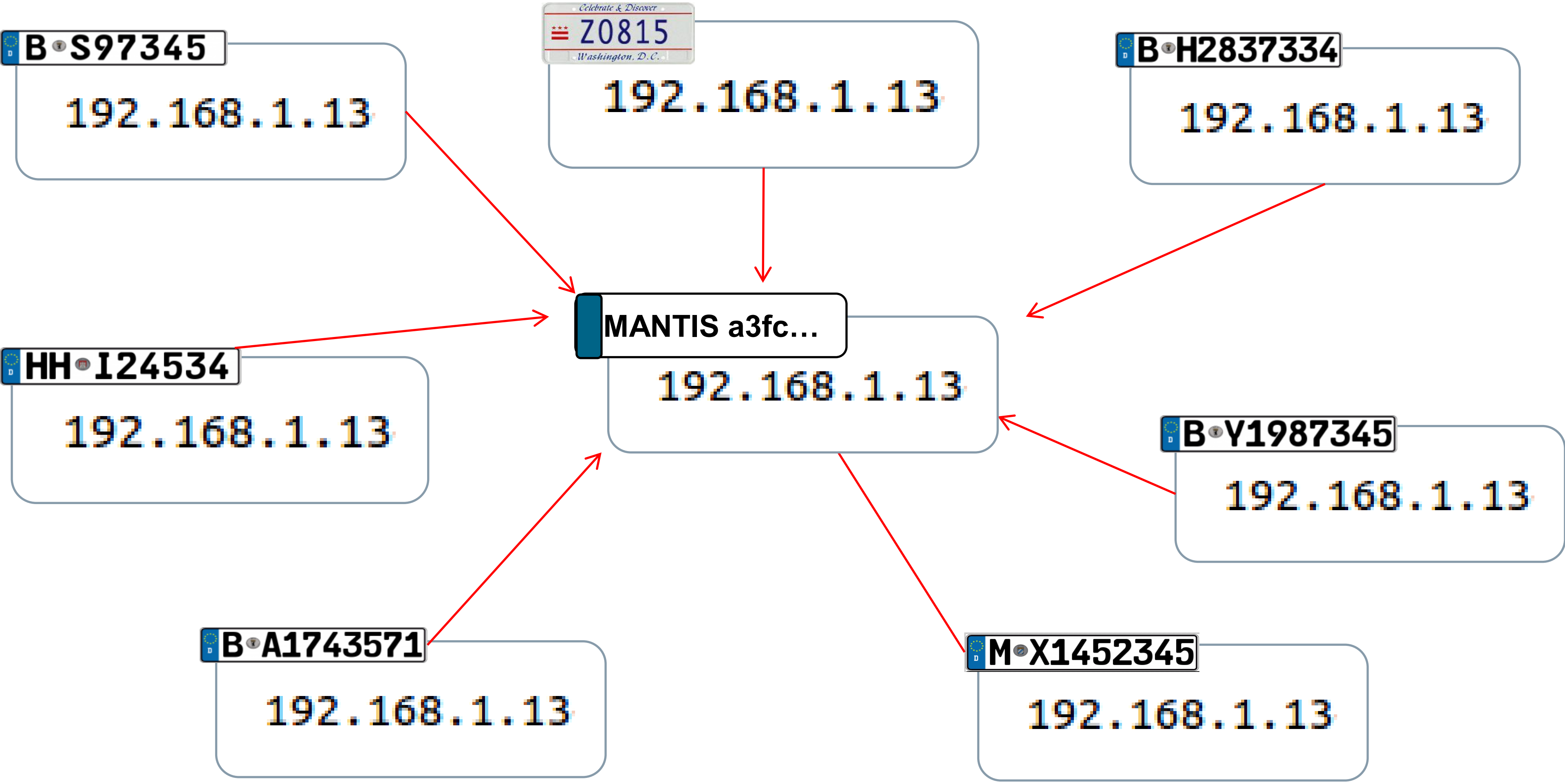
192.168.1.13

192.168.1.13

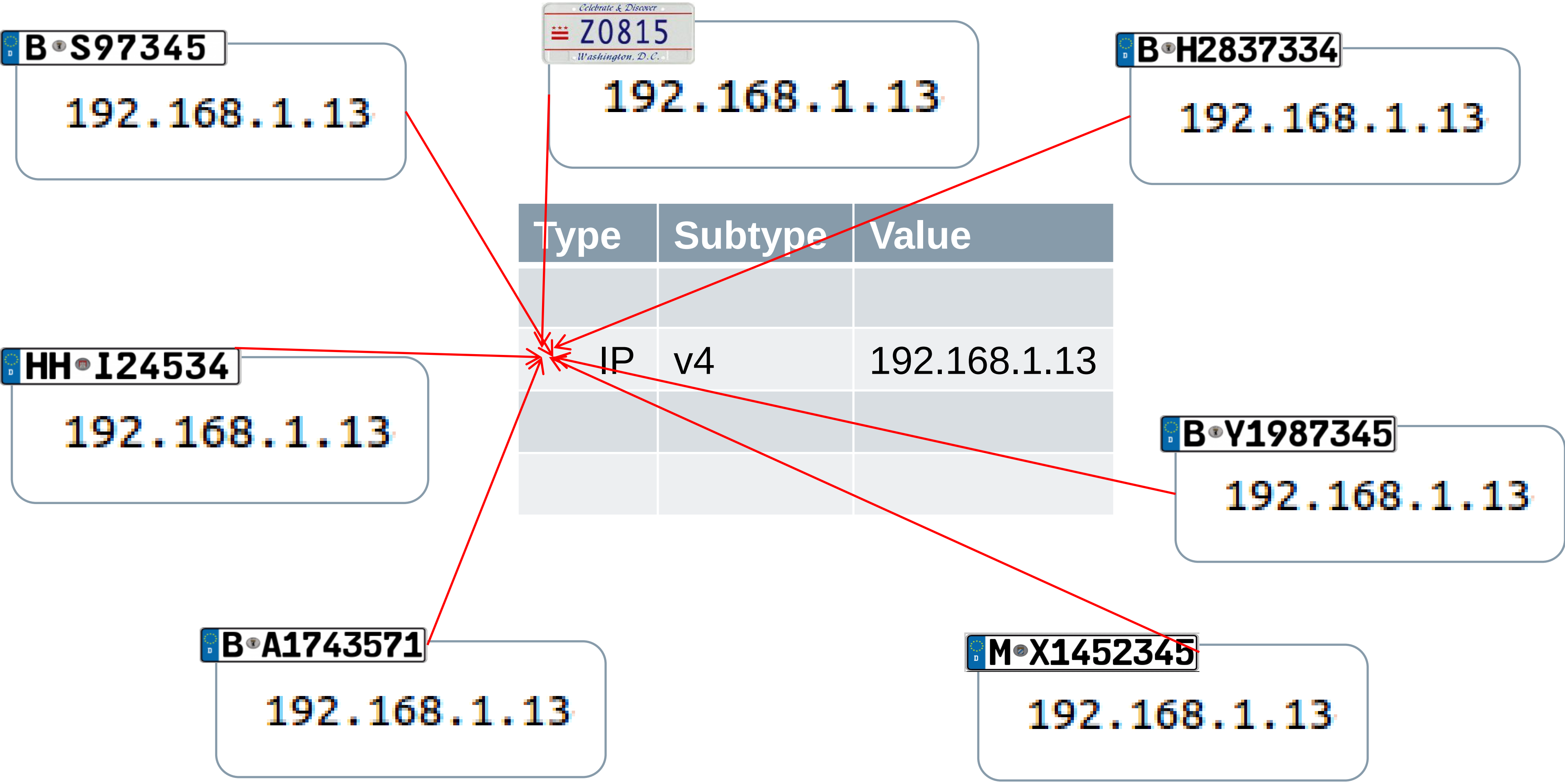
192.168.1.13

How to solve this?
Here, I was lead into TEMPTATION … and succumbed

I could have resisted the temptation and used a CybOX object as canonical representation …

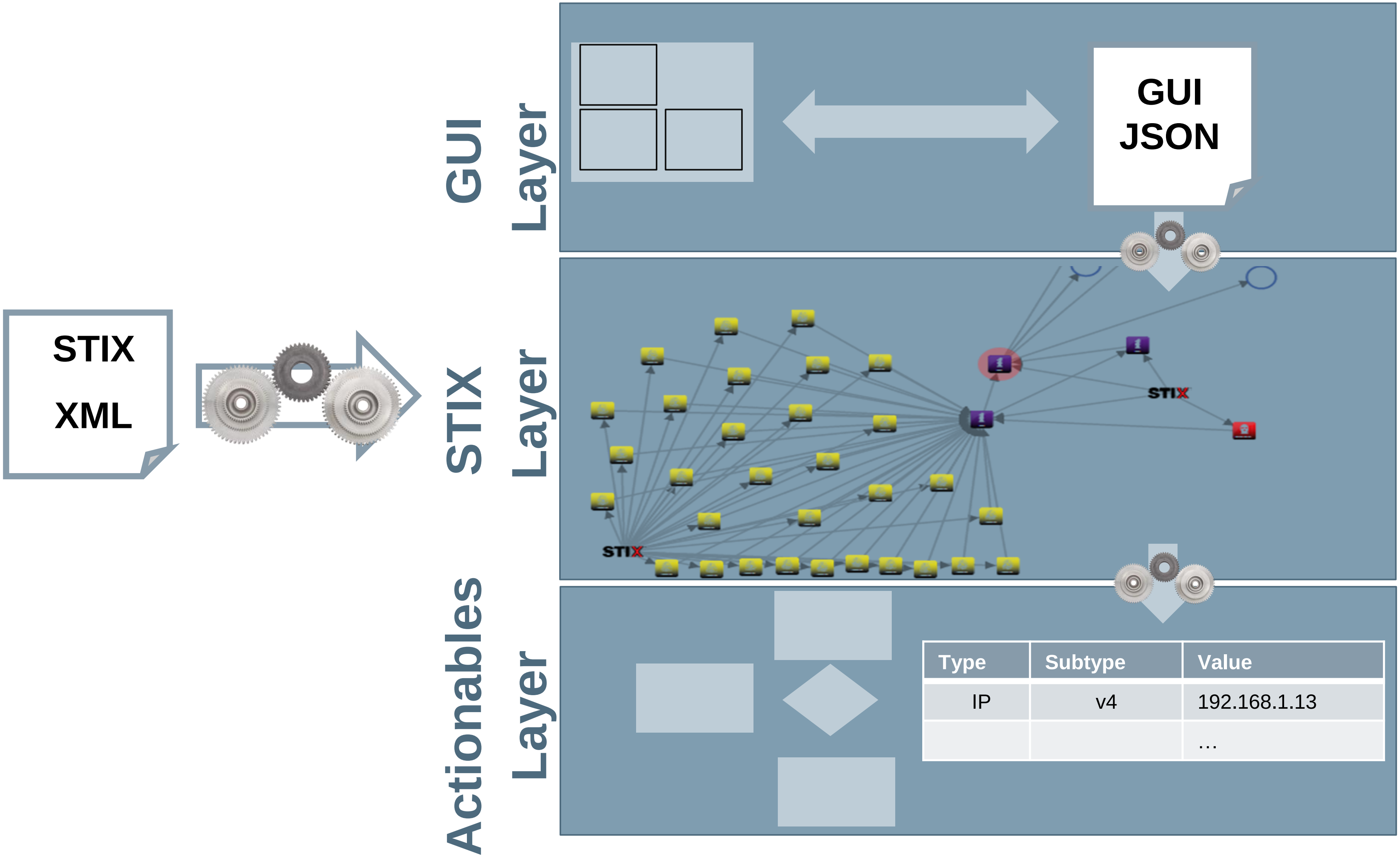


… but I succumbed and created a new database table for „simplistic observables“ …



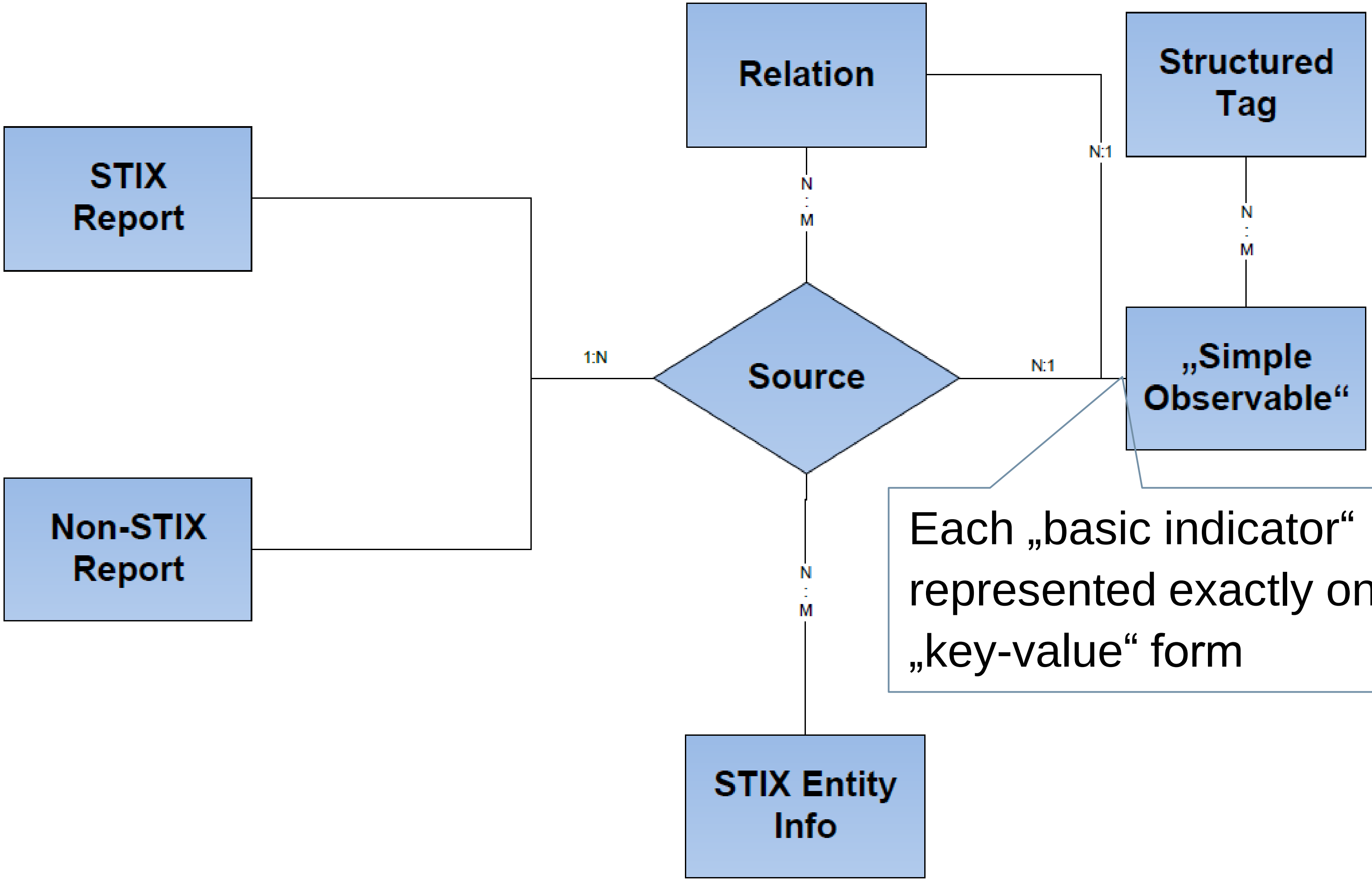
Type	Subtype	Value
IP	v4	192.168.1.13

… and *BANG*, we have a third layer with a new data model!



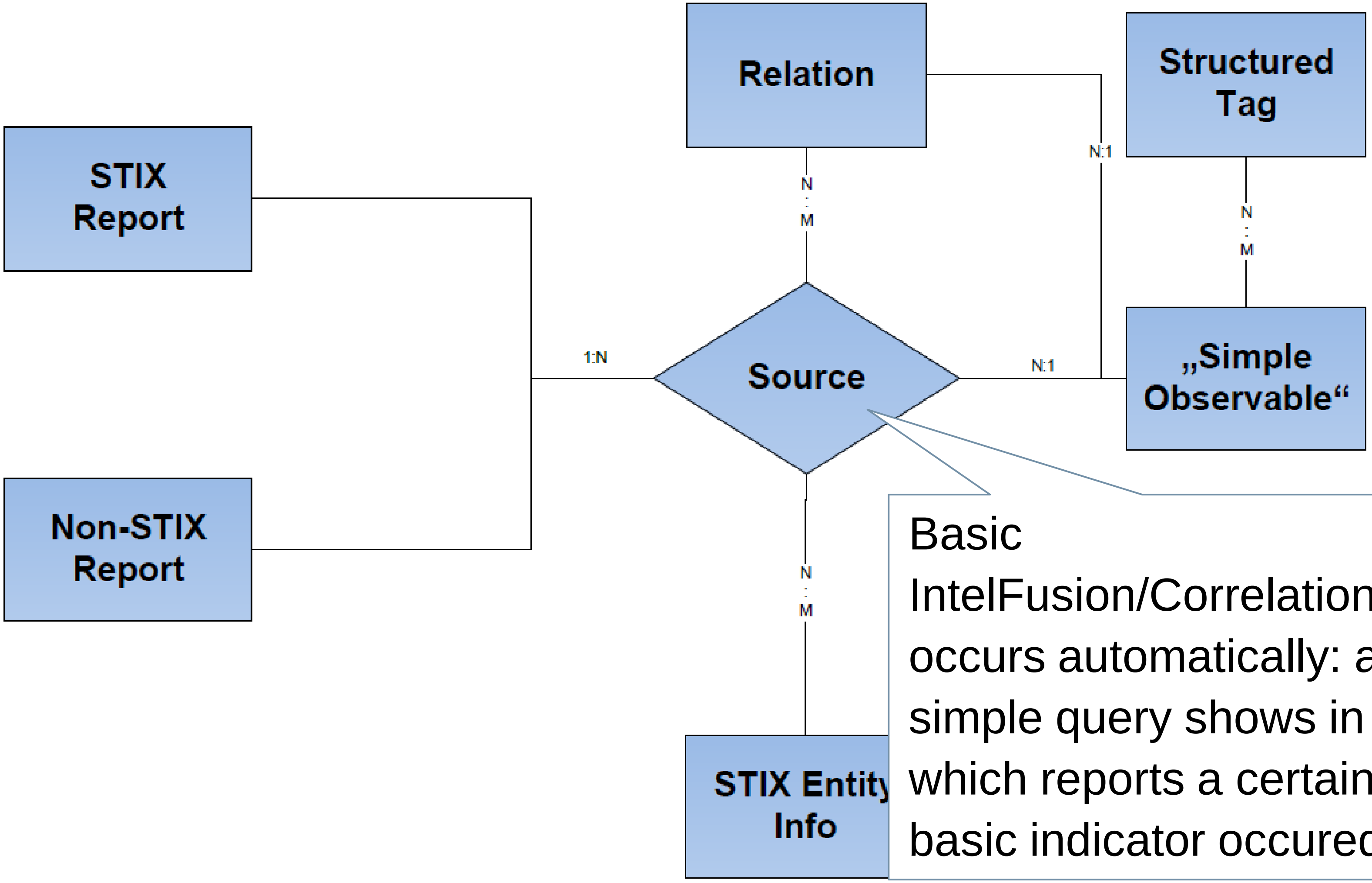
Type	Subtype	Value
IP	v4	192.168.1.13
		…

A closer look at the basics of the MANTIS Actionables Layer



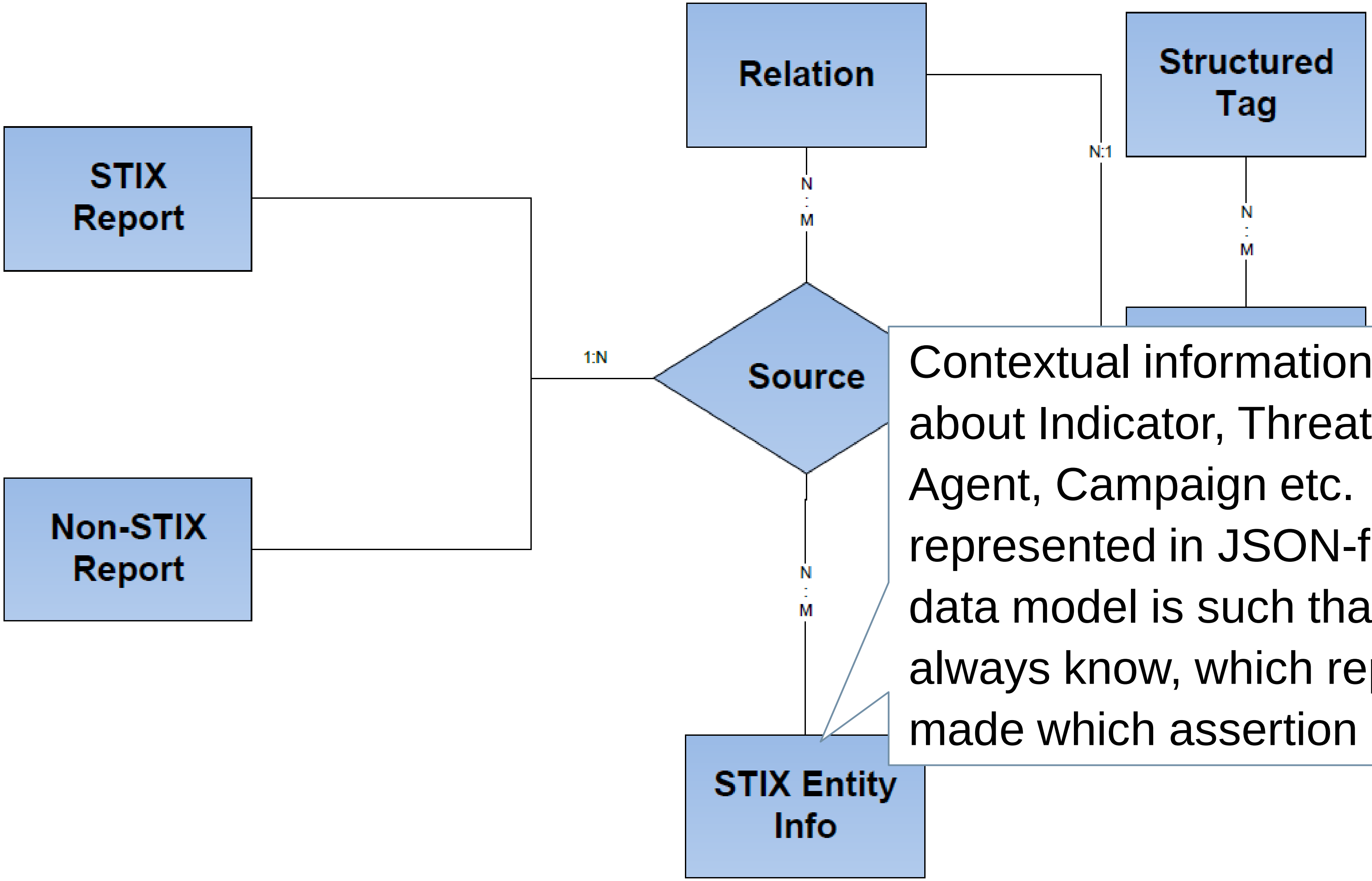
Each „basic indicator“ represented exactly once in „key-value“ form

A closer look at the basics of the MANTIS Actionables Layer



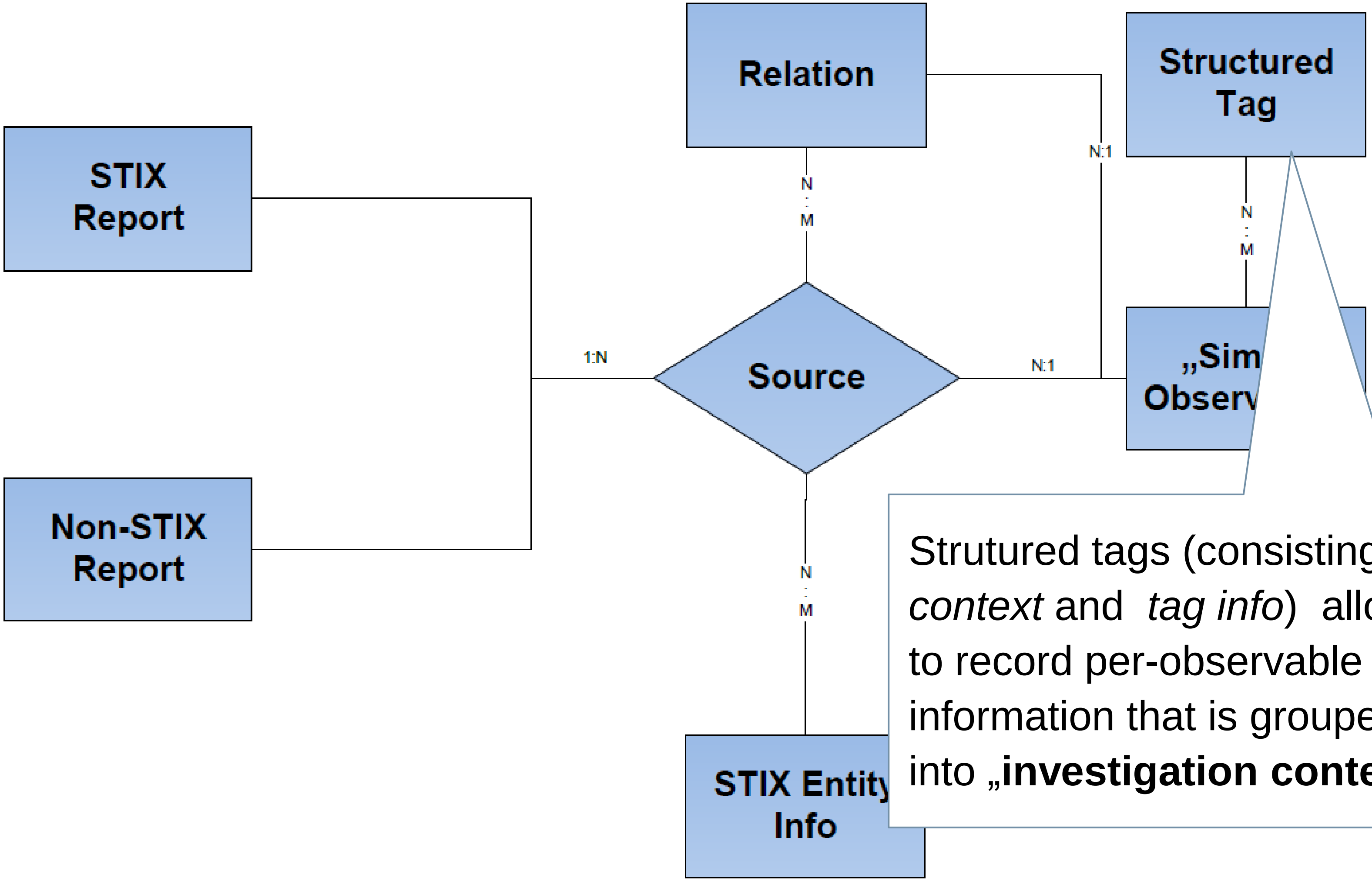
Basic IntelFusion/Correlation occurs automatically: a simple query shows in which reports a certain basic indicator occured

A closer look at the basics of the MANTIS Actionables Layer



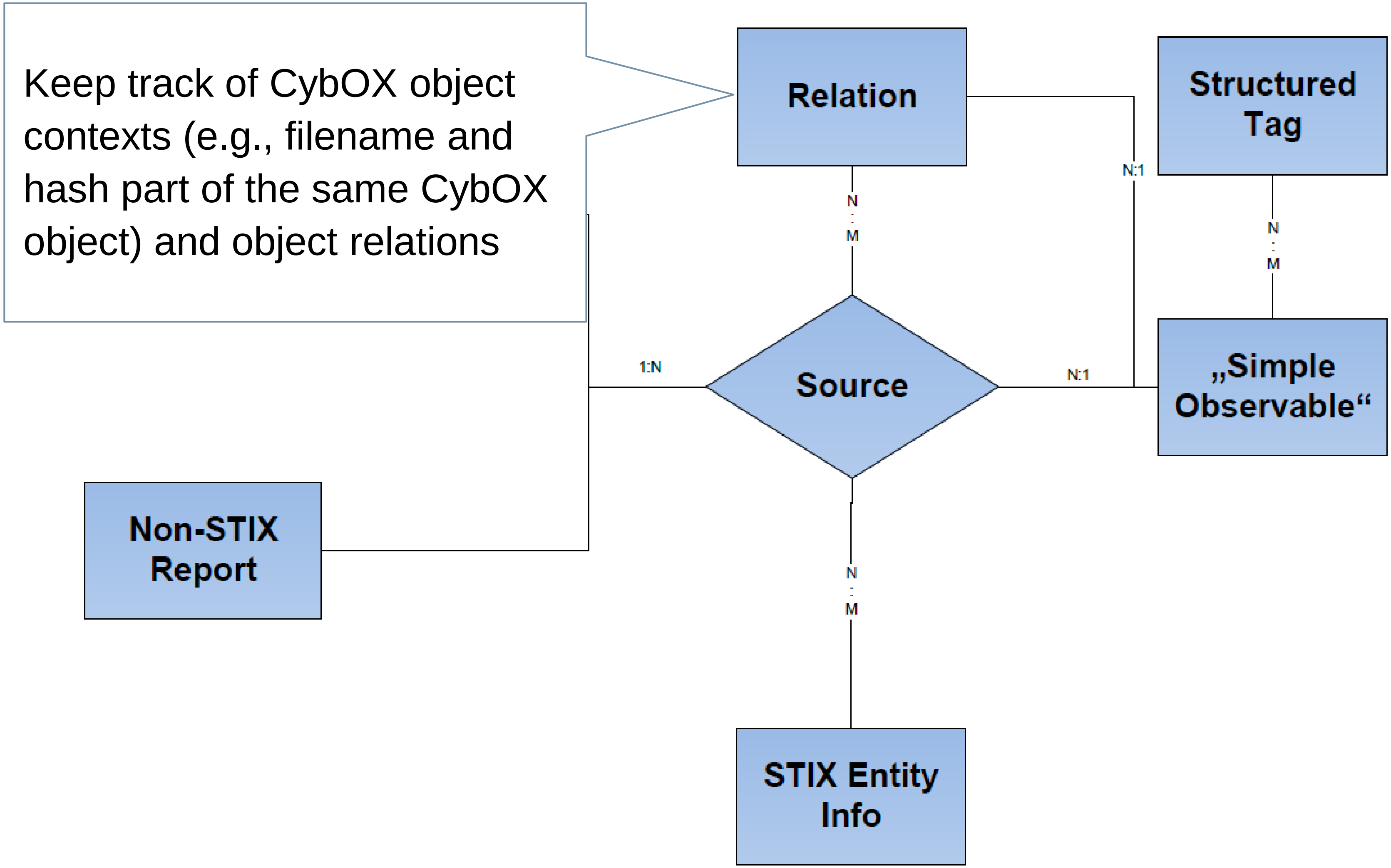
Contextual information about Indicator, Threat Agent, Campaign etc. represented in JSON-form; data model is such that we always know, which report made which assertion …

A closer look at the basics of the MANTIS Actionables Layer



Strutured tags (consisting of *context* and *tag info*) allow us to record per-observable information that is grouped into „**investigation contexts**“

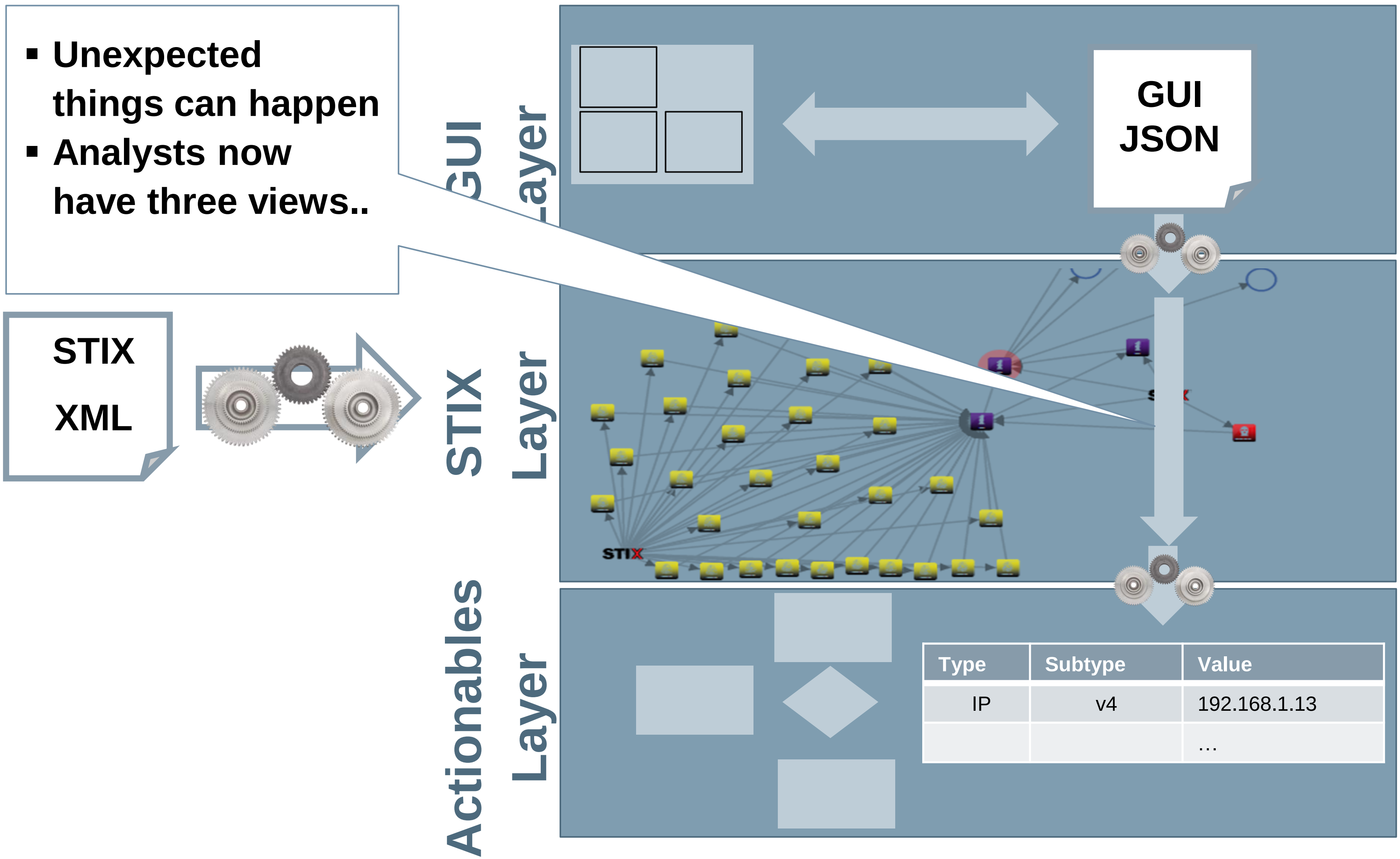
A closer look at the basics of the MANTIS Actionables Layer



Keep track of CybOX object contexts (e.g., filename and hash part of the same CybOX object) and object relations

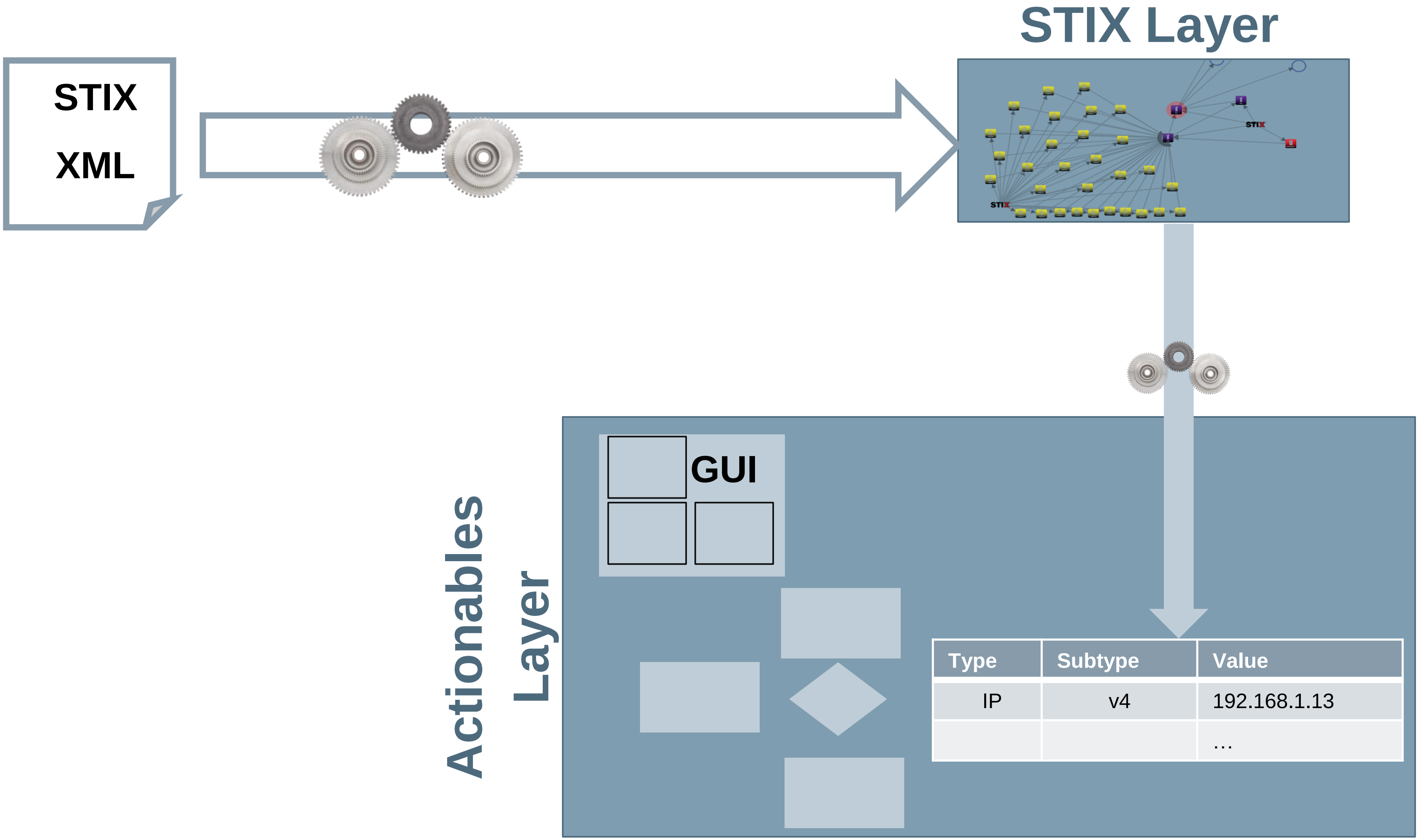
Here is a headache:

- Unexpected things can happen
- Analysts now have three views..



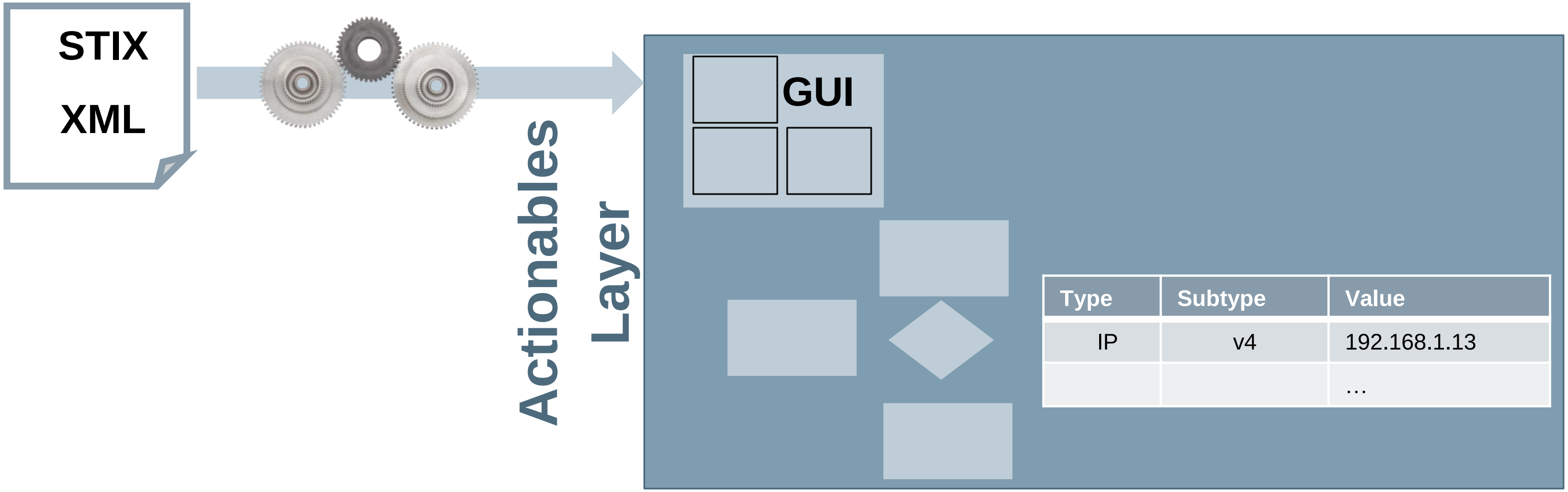
Type	Subtype	Value
IP	v4	192.168.1.13
		…

So we are probably moving towards the following: Midterm:



Type	Subtype	Value
IP	v4	192.168.1.13
		…

So we are probably moving towards the following: Longterm:



Type	Subtype	Value
IP	v4	192.168.1.13
		…

Possible Strategy for long term development: „Mantis in MISP“:

- MISP already has
  - key-value pair representation of basic indicators
  - machine tags
- MISP currently lacks (but has lot‘s of this on the roadmap):
  - per-indicator tagging
  - structured way to represent contextual information and „object containment“ from STIX entities
  - single representation of basic indicators (and thus „fusion/correlation for free“)

Conclusions

- **Using STIX 1.x/CybOX 2.x „directly“ as data model rather than data exchange model is hard:**
  - Requirements for use-case support / templating are likely to lead to a separate GUI layer
  - Chances are that you end up with a second, internal data model (and third layer) that helps you deal with what is really actionable

- **STIX 2.x/CybOX 3.x may make „direct“ usage easier, but still:**
  ~~YOUR USECASE~~ SOLVING YOUR PROBLEM comes FIRST!!!

- **MANTIS is doing more and more based on a data model that**
  - represents „simple observable“ / „basic indicators“ as key value pairs
  - supports basic fusion/correlation „for free“ by deduplicating basic indicators
  - bases the analysts‘ work in „investigation contexts“

- **We are evaluating the possibility of a „Mantis in MISP“ approach**
  - MISP well-established in indicator sharing with broad user base in Europe and excellent code maintenance / further development
  - MISP roadmap looks like „Mantis in MISP“ is a realistic possibility