# pathogen-identified.xml

from:

<element>  
 <path value="Observation.component" />  
 <slicing>

To:

<element id="Observation.component">  
 <path value="Observation.component" />  
 <slicing>

# infection-condition.xml

from:

<element>  
 <path value="Condition.extension" />  
 <slicing>

To:

<element id="Condition.extension">  
 <path value="Condition.extension" />  
 <slicing>

# hai-surgical-site-infection-report-ssi.xml

from:

<element>  
 <path value="Composition.section" />  
 <slicing>

To:

<element id="Composition.section">  
 <path value="Composition.section" />  
 <slicing>

# hai-procedure-denominator-report.xml

from

<element>  
 <path  
 value="Composition.section" />  
 <slicing>

to

# All value set files:

Remove meta element

# Healthcare\_Associated\_Infection\_Reports\_FHIR\_SSI\_Prototype.json

Added dependency list:

"dependencyList": [{  
 "name": "USCore",  
 "location": "http://hl7.org/fhir/us/core/"   
 },{  
 "name": "ccda",  
 "location": "http://build.fhir.org/ig/HL7/ccda-on-fhir/"  
 }],

Changed specification (not sure if this was really necessary or not!)

From:

"specification": "http://hl7-fhir.github.io/"

To

"specification": "http://hl7.org/fhir/STU3/"

# ImplementationGuide\_3347.xml

# Changed order of resources so that included resources come first (patient etc)., then compositions in order of parent/child/etc.

# Instance-template-sd.html

Copied file from C-CDA on FHIR because was getting a ton of errors on the JSON template tab – C-CDA on FHIR has removed the XML template and the JSON template tabs (doing this brought the build errors down from 3097 to 3!)

# hai-surgical-site-infection-report-ssi

Got differential does not have a slice: Composition.section.entry in profile <http://hl7.org/fhir/us/hai/StructureDefinition/hai-surgical-site-infection-report-ssi>

So did the following (this is something that always has to be done manually, Trifolia can’t do it):

<!-- SG: Commented out and moved pathogen-identified below to make it a simple choice rather than a slice -->  
 <!--<element id="Composition.section:section4.entry">  
 <path value="Composition.section.entry" />  
 <short value="entry" />  
 <definition value="MAY contain [0..3] Pathogen Identified (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/pathogen-identified) (CONF:3347-125)." />  
 <min value="0" />  
 <max value="3" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/pathogen-identified" />  
 </type>  
 </element>-->  
 <element id="Composition.section:section4.entry">  
 <path value="Composition.section.entry" />  
 <short value="entry" />  
 <definition value="MAY contain zero or one [0..1] MDRO/CDI Observation (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/mdro-cdi-observation) (CONF:3347-136)." />  
 <min value="0" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/mdro-cdi-observation" />  
 </type>  
 <!-- SG: Moved from commented out element above -->  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/pathogen-identified" />  
 </type>  
 </element>

# hai-procedure-denominator-report.xml

Got differential does not have a slice: Composition.section.entry in profile

<!-- SG: Commented out and moved asa-class below to make it a simple choice rather than a slice -->  
 <!--<element id="Composition.section:section3.entry">  
 <path value="Composition.section.entry" />  
 <short value="entry" />  
 <definition  
 value="SHALL contain exactly one [1..1] ASA Class (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/asa-class) (CONF:3347-218)." />  
 <min value="1" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/asa-class" />  
 </type>  
 </element>-->  
 <element id="Composition.section:section3.entry">  
 <path value="Composition.section.entry" />  
 <short value="entry" />  
 <definition  
 value="SHALL contain exactly one [1..1] Diabetes Melitus (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/diabetes-melitus) (CONF:3347-219)." />  
 <min value="1" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/diabetes-melitus" />  
 </type>  
 <!-- SG: Moved from commented out element above -->  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/asa-class" />  
 </type>  
 </element>

# procedure-details.xml

Not all <element id’s are unique – fixed this. The Schematron doesn’t catch this- it might be a problem in other places

<element id="partOf1">  
 <path value="Procedure.partOf" />  
 <short value="partOf" />  
 <definition  
 value="MAY contain zero or one [0..1] Anesthesia Delivered (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/anesthesia-delivered) (CONF:3347-185)." />  
 <min value="0" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/anesthesia-delivered" />  
 </type>  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/closure-technique" />  
 </type>  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/spinal-fusion-level" />  
 </type>  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/revision-associated-with-prior-infection" />  
 </type>  
 </element>  
 <!-- SG: Commented out and moved choices up to make it a simple choice rather than a slice -->  
 <!--<element id="partOf2">  
 <path value="Procedure.partOf" />  
 <short value="partOf" />  
 <definition  
 value="MAY contain zero or one [0..1] Closure Technique (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/closure-technique) (CONF:3347-208)." />  
 <min value="0" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/closure-technique" />  
 </type>  
 </element>-->  
 <!--<element id="partOf3">  
 <path value="Procedure.partOf" />  
 <short value="partOf" />  
 <definition  
 value="MAY contain zero or one [0..1] Spinal Fusion Level (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/spinal-fusion-level) (CONF:3347-209)." />  
 <min value="0" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/spinal-fusion-level" />  
 </type>  
 </element>-->  
 <!--<element id="partOf4">  
 <path value="Procedure.partOf" />  
 <short value="partOf" />  
 <definition  
 value="MAY contain zero or one [0..1] Revision Associated with Prior Infection Observation (identifier: http://hl7.org/fhir/us/hai/StructureDefinition/revision-associated-with-prior-infection) (CONF:3347-210)." />  
 <min value="0" />  
 <max value="1" />  
 <type>  
 <code value="Reference" />  
 <targetProfile value="http://hl7.org/fhir/us/hai/StructureDefinition/revision-associated-with-prior-infection" />  
 </type>  
 </element>-->

# procedure-risk-factors.xml

differential issue

FHIR Validation Changes (nothing to do with Trifolia)

Observation.sch

Change:

<sch:assert test="not(exists(\*[starts-with(local-name(.), 'value')])) or not(count(for $coding in code/coding return parent::\*/component/code/coding[code/@value=$coding/code/@value and system/@value=$coding/system/@value])=0)">obs-7: If code is the same as a component code then the value element associated with the code SHALL NOT be present</sch:assert>

To: (have raised GForge Issue)

<sch:assert test="not(exists(f:\*[starts-with(local-name(.), 'value')])) or count(for $coding in f:code/f:coding return self::\*/f:component/f:code/f:coding[f:code/@value=$coding/f:code/@value and f:system/@value=$coding/f:system/@value])=0">obs-7: If code is the same as a component code then the value element associated with the code SHALL NOT be present</sch:assert>

Questionnaire.sch

Change:

<sch:rule context="f:Questionnaire/f:item/f:enableWhen">  
 <sch:assert test="count(f:\*[starts-with(local-name(.), 'answer')]|self::f:hasAnswer) = 1">que-7: enableWhen must contain either a 'answer' or a 'hasAnswer' element</sch:assert>  
 </sch:rule>

To (Sean raised GForge issue):

<sch:rule context="f:Questionnaire/f:item/f:enableWhen">  
 <sch:assert test="count(f:\*[starts-with(local-name(.), 'answer')]|f:hasAnswer) = 1">que-7: enableWhen must contain either a 'answer' or a 'hasAnswer' element</sch:assert>  
 </sch:rule>

# Pages/\_includes files

Manually updated all these files