PSI-XML schema update

1) Update BibRef element

a) Context

The *BibRef* element refers to a publication. It currently allows either a *Xref* element (to describe pubmed primary reference if it exists) **OR** an *AttributeList* element (to describe publication details such as publication title, authors, journal, publication date, ...). So when we want to export **both** pubmed primary reference and publication details, we add the pubmed primary reference in *BibRef* and the publication details attributes in the *AttributeList* of *ExperimentDescription.*

Example:

```
−<experimentDescription id="1035263">
  +<names></names>
  −<bibref>
    −<xref>
       <primaryRef refTypeAc="MI:0358" refType="primary-reference" id="10049357" dbAc="MI:0446" db="pubmed"/>
     </xref>
   </bibref>
  +<xref></xref>
  +<hostOrganismList></hostOrganismList>
  +<interactionDetectionMethod></interactionDetectionMethod>
  +<participantIdentificationMethod></participantIdentificationMethod>
  −<attributeList>
    −<attribute nameAc="MI:0636" name="author-list">
       Nomura T., Khan M.M., Kaul S.C., Dong H.D., Wadhwa R., Colmenares C., Kohno I., Ishii S.
     </attribute>
    −<attribute nameAc="MI:0633" name="data-processing">
       Origin of SnoN/SKIL inferred from sequence length of the protein, of mSin3a according author name Ishii S. which is associated with the RIKEN institute
       having submitted human Sin3a. Exp using SMRT (fig 1C) omitted as the protein cannot be mapped to UniProt.
     </attribute>
     <attribute nameAc="MI:0634" name="contact-email">sishii@rtc.riken.go.jp</attribute>
     <attribute nameAc="MI:0885" name="journal">Genes Dev. (0890-9369)</attribute>
     <attribute nameAc="MI:0886" name="publication year">1999</attribute>
     <attribute nameAc="MI:0955" name="curation depth">mimix curation</attribute>
   </attributeList>
</experimentDescription>
```

b) Propositions

We propose to update BibRef element to accept **both *Xref* and *AttributeList*** so the publication can be entirely described in BibRef.

Example:

```
-<experimentDescription id="1035263">
 +<names></names>
 -<bibref>
  -<xref>
     <primaryRef refTypeAc="MI:0358" refType="primary-reference" id="10049357" dbAc="MI:0446" db="pubmed"/>
   </xref>
  -<attributeList>
    -<attribute nameAc="MI:0636" name="author-list">
       Nomura T., Khan M.M., Kaul S.C., Dong H.D., Wadhwa R., Colmenares C., Kohno I., Ishii S.
     </attribute>
     <attribute nameAc="MI:0634" name="contact-email">sishii@rtc.riken.go.jp</attribute>
     <attribute nameAc="MI:0885" name="journal">Genes Dev. (0890-9369)</attribute>
     <attribute nameAc="MI:0886" name="publication year">1999</attribute>
     <attribute nameAc="MI:0955" name="curation depth">mimix curation</attribute>
   </attributeList>
 </bibref>
 +<xref></xref>
 +<hostOrganismList></hostOrganismList>
 +<interactionDetectionMethod></interactionDetectionMethod>
 +<participantIdentificationMethod></participantIdentificationMethod>
 -<attributeList>
  -<attribute nameAc="MI:0633" name="data-processing">
     Origin of SnoN/SKIL inferred from sequence length of the protein, of mSin3a according author name Ishii S. which is associated with the RIKEN institute
     having submitted human Sin3a. Exp using SMRT (fig 1C) omitted as the protein cannot be mapped to UniProt.
   </attribute>
 </attributeList>
</experimentDescription>
```

The XSD changes can be found here.

2) Optional experimentRef in parameter

a) Context

The current *parameter* element requires to give an *experimentRef*. The *experimentRef* element should be optional the same way the *experimentRefList* is optional in the *confidence* element. If the interaction has only one experiment, it is obvious that the parameter comes from this experiment and we should not have to add a supplementary *experimentRef* element to the parameter.

b) Propositions

Therefore, we propose to make the *experimentRef* optional in the *parameter* element.

The XSD changes can be found here.

3) Update positionType and intervalType for feature ranges

a) Context

The current feature range positions have a type ‘*unsignedLong*’ which means that features can only have positive range positions. However, we now need to describe promotor regions which are usually represented as negative positions.

b) Propositions

Therefore, we propose to update positionType and intervalType from ‘*unsignedLong*’ to *‘long’* to enable negative positions for the feature ranges.

The XSD changes can be found here.

4) Adding resulting sequence for mutations

a) Context

The position and effect of a mutation can be systematically captured using the *FeatureRange* positions and the *FeatureType* element. However, it is currently not well defined how to capture the actual sequence change (IntAct currently encodes the sequence change in the feature shortLabel element but it is not standard as shown in the example below).

Example:

```
−<feature id="180661">
  −<names>
     <shortLabel>his33arg ser53pro</shortLabel>
     <fullName>his33arg ser53pro lys72arg</fullName>
   </names>
  +<xref></xref>
  −<featureType>
    −<names>
       <shortLabel>mutation decreasing</shortLabel>
       <fullName>mutation decreasing interaction</fullName>
       <alias type="go synonym" typeAc="MI:0303">hotspot</alias>
     </names>
    +<xref></xref>
   </featureType>
  +<featureDetectionMethod></featureDetectionMethod>
  −<featureRangeList>
    −<featureRange>
      +<startStatus></startStatus>
       <begin position="33"/>
      +<endStatus></endStatus>
       <end position="33"/>
       <isLink>false</isLink>
     </featureRange>
    −<featureRange>
      +<startStatus></startStatus>
       <begin position="53"/>
      +<endStatus></endStatus>
       <end position="53"/>
       <isLink>false</isLink>
     </featureRange>
    −<featureRange>
      +<startStatus></startStatus>
       <begin position="72"/>
      +<endStatus></endStatus>
       <end position="72"/>
       <isLink>false</isLink>
     </featureRange>
   </featureRangeList>
```

b) Propositions

We propose to add a new element named *resultingSequence* at the

level of the *featureRange* element. The *resultingSequence* element should not be at the *feature* level because a feature can have several ranges and the *resultingSequence* element is directly related to the range positions. The *resultingSequence* element would contain some information about the sequence change.

The XSD changes can be found here.

b.1) Proposal 1

In the *resultingSequence* element, we would need a *sequence* element which would be required and would describe the resulting sequence and a *xref* element which would be optional and could be used to add external cross references such as dbSNP cross references.

Notes: *The original sequence is not explicit and can be extracted from interactor sequence and range positions. However, if the sequence is not exported or if we don't provide regular protein updates, we may infer wrong mutations because we don't explicitly say which amino acid was mutated.*

Example:

```
−<featureRangeList>
  −<featureRange>
    +<startStatus></startStatus>
      <begin position="33"/>
    +<endStatus></endStatus>
      <end position="33"/>
      <isLink>false</isLink>
    −<resultingSequence>
        <sequence>R</sequence>
      −<xref>
          <primaryRef refTypeAc="MI:0356" refType="identity" id="rs1042522" db="dbSNP"/>
        </xref>
      </resultingSequence>
    </featureRange>
  −<featureRange>
    +<startStatus></startStatus>
      <begin position="53"/>
    +<endStatus></endStatus>
      <end position="53"/>
      <isLink>false</isLink>
    −<resultingSequence>
        <sequence>P</sequence>
      </resultingSequence>
    </featureRange>
  −<featureRange>
    +<startStatus></startStatus>
      <begin position="72"/>
    +<endStatus></endStatus>
      <end position="72"/>
      <isLink>false</isLink>
    −<resultingSequence>
        <sequence>R</sequence>
      −<xref>
          <primaryRef refTypeAc="MI:0356" refType="identity" id="rs1042524" db="dbSNP"/>
        </xref>
      </resultingSequence>
    </featureRange>
  </featureRangeList>
```

b.2) Proposal 2

In the *resultingSequence* element, we would need a *originalSequence* element which would be required and would describe the original sequence, a *newSequence* element which would be required and would describe the resulting sequence and a *xref* element which would be

optional and could be used to add external cross references such as dbSNP cross references.

Notes: *The original sequence is explicit so we know exactly which amino acid is mutated even if the interactor sequence is not up to date or is not exported in the XML. However, we duplicate the information about the original sequence as we can extract it from the range positions and interactor sequence.*

Example:

```
−<featureRange>
  +<startStatus></startStatus>
    <begin position="33"/>
  +<endStatus></endStatus>
    <end position="33"/>
    <isLink>false</isLink>
  −<resultingSequence>
      <originalSequence>H</originalSequence>
      <newSequence>R</newSequence>
    −<xref>
        <primaryRef refTypeAc="MI:0356" refType="identity" id="rs1042522" db="dbSNP"/>
      </xref>
    </resultingSequence>
  </featureRange>
−<featureRange>
  +<startStatus></startStatus>
    <begin position="53"/>
  +<endStatus></endStatus>
    <end position="53"/>
    <isLink>false</isLink>
  −<resultingSequence>
      <originalSequence>S</originalSequence>
      <newSequence>P</newSequence>
    </resultingSequence>
  </featureRange>
−<featureRange>
  +<startStatus></startStatus>
    <begin position="72"/>
  +<endStatus></endStatus>
    <end position="72"/>
    <isLink>false</isLink>
  −<resultingSequence>
      <originalSequence>K</originalSequence>
      <newSequence>R</newSequence>
    −<xref>
        <primaryRef refTypeAc="MI:0356" refType="identity" id="rs1042524" db="dbSNP"/>
      </xref>
    </resultingSequence>
```

5) Adding stoichiometry to participant

a) Context

There is currently no standard way to export stoichiometry information about a participant in an interaction. Some databases such as IntAct export this information as an attribute of the participant (see example below) but it is really specific to each database.

Example:

```
−<participant id="16701">
  +<names></names>
  +<xref></xref>
    <interactorRef>16699</interactorRef>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
  +<featureList></featureList>
  −<attributeList>
      <attribute nameAc="MI:0612" name="comment">Stoichiometry: 2.0</attribute>
    </attributeList>
  </participant>
```

b) Propositions

The possible XSD changes are available here.

b.1) Proposal 1

We propose to update *Participant* element to add an optional *stoichiometry* attribute which would describe the stoichiometry of this participant as a decimal value. We could also describe stoichiometry ranges by adding an optional *minStoichiometry* attribute and an optional *maxStoichiometry* attribute.

Notes: *If we add these three optional attributes, we cannot enforce at the schema level some rules to avoid some inconsistencies such as:*
- We cannot have both stoichiometry attribute and minStoichiometry/maxStoichiometry attribute in the same participant element
- If someone describes of stoichiometry range, both minStoichiometry and maxStoichiometry are required

Example with stoichiometry attribute :

```
−<participant id="16701" stoichiometry="2">
  +<names></names>
  +<xref></xref>
    <interactorRef>16699</interactorRef>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
  +<featureList></featureList>
  </participant>
```

Example with stoichiometry range attributes :

```
−<participant id="16701" minStoichiometry="1" maxStoichiometry="3">
  +<names></names>
  +<xref></xref>
    <interactorRef>16699</interactorRef>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
  +<featureList></featureList>
  </participant>
```

b.2) Proposal 2

We propose to update *Participant* element to add an optional XSD choice sub-element which provides a choice between a *stoichiometry* element which would describe the stoichiometry for this participant or a *stoichiometryRange* element which would describe a stoichiometry range for this participant. If the *stoichiometry* element is chosen, a *value* attribute

is required to describe the stoichiometry as a decimal value. If the *stoichiometryRange* element is chosen, both *minValue* and *maxValue* attributes are required to describe the stoichiometry range as decimal values.

Notes: *If we add these two optional nodes, we can enforce some consistency in the schema when we describe a stoichiometry where only a single value is expected or a stoichiometry range where both minValue and maxValue should be provided.*

Example with stoichiometry element:

```
−<participant id="16701">
  +<names></names>
  +<xref></xref>
    <interactorRef>16699</interactorRef>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
  +<featureList></featureList>
    <stoichiometry value="2"/>
  </participant>
```

Example with stoichiometry range element:

```
−<participant id="16701">
  +<names></names>
  +<xref></xref>
    <interactorRef>16699</interactorRef>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
  +<featureList></featureList>
    <stoichiometryRange minValue="1" maxValue="2"/>
  </participant>
```

6) Adding a featureDetectionMethodList in the feature element

a) Context

The current schema can only describe one feature detection method per feature.

Example with mutation analysis:

```
−<feature id="180661">
  +<names></names>
  +<xref></xref>
  +<featureType></featureType>
  −<featureDetectionMethod>
    −<names>
       <shortLabel>mutation analysis</shortLabel>
       <fullName>mutation analysis</fullName>
     </names>
    −<xref>
       <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:0074" dbAc="MI:0488" db="psi-mi"/>
       <secondaryRef refTypeAc="MI:0356" refType="identity" id="EBI-456810" dbAc="MI:0469" db="intact"/>
       <secondaryRef refTypeAc="MI:0358" refType="primary-reference" id="14755292" dbAc="MI:0446" db="pubmed"/>
     </xref>
   </featureDetectionMethod>
  +<featureRangeList></featureRangeList>
  </feature>
```

However, It may happen that a feature represents a PTM which has been identified by mass spec and then confirmed by western blot and we would like to export both of them as feature detection methods for this feature.

b) Propositions

We propose to replace the *featureDetectionMethod* element with a *featureDetectionMethodList* element which is optional and can contains one to several *featureDetectionMethod* elements.

The XSD changes are available here.

Notes: *This change will not be backward compatible with the 2.5 schema.*

Example:

```
−<feature id="180661">
  +<names></names>
  +<xref></xref>
  +<featureType></featureType>
  −<featureDetectionMethodList>
    −<featureDetectionMethod>
      −<names>
          <shortLabel>modified residue ms</shortLabel>
          <fullName>mass detection of residue modification</fullName>
        </names>
      −<xref>
          <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:0068" dbAc="MI:0488" db="psi-mi"/>
        </xref>
      </featureDetectionMethod>
    −<featureDetectionMethod>
      −<names>
          <shortLabel>western blot</shortLabel>
          <fullName>western blot</fullName>
        </names>
      −<xref>
          <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:0113" dbAc="MI:0488" db="psi-mi"/>
        </xref>
      </featureDetectionMethod>
    </featureDetectionMethodList>
  +<featureRangeList></featureRangeList>
  </feature>
```

7) Update feature element to capture interaction dependency and interaction effect

a) Context

We want to capture the fact that some features have an interaction dependency and/or interaction effect (with the interaction that reports the feature):

- **prerequisite-ptm**: Post translational modification required for an interaction to occur.
- **ptm decreasing an interaction:** Post translational modification on a protein observed to decrease the strength or rate of an interaction.
- **ptm increasing an interaction:** Post translational modification on a protein observed to increase the strength or rate of an interaction.
- **ptm disrupting an interaction:** Post translational modification on a protein observed to disrupt the strength or rate of an interaction.
- **resulting-ptm**: Post translational modification occurs subsequently to an interaction.
- hidden binding site that is revealed after the interaction occurs, …

Currently, this information is stored as an attribute of a feature. Each possible interaction dependency/effect is a CV term in the PSI-MI ontology (children of feature attribute names and observed ptm).

Example:

```
−<feature id="52">
  −<featureType>
    −<names>
       <shortLabel>phosphorylated residue</shortLabel>
     </names>
    −<xref>
       <primaryRef refType="identity" refTypeAc="MI:0356" db="psi-mod" dbAc="MI:0897" id="MOD:00696"/>
     </xref>
   </featureType>
  −<featureRangeList>
    −<featureRange>
     +<startStatus></startStatus>
       <begin position="826"/>
     +<endStatus></endStatus>
       <end position="826"/>
     </featureRange>
   </featureRangeList>
  −<attributeList>
     <attribute name="resulting-ptm" nameAc="MI:0639"/>
   </attributeList>
 </feature>
```

b) Propositions

The XSD changes are available here.

b.1) Proposal 1

We propose to replace the *featureType* element in the *feature* element with a *featureTypeList* element which is optional and can contain one to many *featureType* elements.

Notes: *This change will not be backward compatible with the 2.5 schema. In addition to that, we usually don't want more than two feature types and resulting-ptm is not a simple feature type, it is an interaction dependency type. Having a featureTypeList could also bring some inconsistency (ex: resulting-ptm cannot be associated with prerequisite-ptm).*

Example:

```
−<feature id="52">
  −<featureTypeList>
    −<featureType>
      −<names>
          <shortLabel>phosphorylated residue</shortLabel>
        </names>
      −<xref>
          <primaryRef refType="identity" refTypeAc="MI:0356" db="psi-mod" dbAc="MI:0897" id="MOD:00696"/>
        </xref>
      </featureType>
    −<featureType>
      −<names>
          <shortLabel>resulting-ptm</shortLabel>
        </names>
      −<xref>
          <primaryRef refType="identity" refTypeAc="MI:0356" db="psi-mi" dbAc="MI:0488" id="MI:0639"/>
        </xref>
      </featureType>
    </featureTypeList>
  +<featureRangeList></featureRangeList>
  </feature>
```

b.2) Proposal 2

We propose to add an optional *interactionDependencyType* (or *featureInteractionDependencyType*) and *interactionEffectType* (or *featureInteractionEffectType*) elements which would be CvType elements.

Notes: *This change is backward compatible with the 2.5 schema. It would enforce that we want only one interaction dependency for this feature (ex: resulting-ptm) and that is slightly different from the feature type which describes the feature (ex: phosphorylated residue).*

Example:

```
−<feature id="52">
  −<featureType>
    −<names>
       <shortLabel>phosphorylated residue</shortLabel>
     </names>
    −<xref>
       <primaryRef refType="identity" refTypeAc="MI:0356" db="psi-mod" dbAc="MI:0897" id="MOD:00696"/>
     </xref>
   </featureType>
  +<featureRangeList></featureRangeList>
  −<interactionDependencyType>
    −<names>
       <shortLabel>resulting-ptm</shortLabel>
     </names>
    −<xref>
       <primaryRef refType="identity" refTypeAc="MI:0356" db="psi-mi" dbAc="MI:0488" id="MI:0639"/>
     </xref>
   </interactionDependencyType>
 </feature>
```

8) Interactor set for complexes and APMS experiments

a) Context

We want to represent a set of interactors as a unique participant in an interaction. This is important for complexes where we we don't know exactly which interactor interacts but we have a possible set of interactors that could interact. It could also be used when we have a uniprot gene demerge and we are not sure which gene was used so we could provide all the possible uniprot entries as an interactor set.

It is also possible that within a set of interactor candidates which do have different sequences, we may want to add specific information such as binding site information. In this case, we are not just talking about using an interactor set in a unique participant but also having a participant set which would be an extension of the participant and used as a same participant entity in the interaction.

Currently the schema does not allow to describe a set of interactors/participants as participant of an interaction.

This issue was discussed during the PSI meeting 2013 (Liverpool) and the first proposition was to add an optional *interactorGroup* element in the *interactionList* that would be used for both complexes and interactor sets. However, unlike complexes, the interactors in a set of interactors are not interacting with each other and therefore are not an interaction, so it would be quite messy to mix it up with interactions in the *interactionList*.

b) Propositions

The XSD changes are available here.

b.1) Proposal 1

As a result, we rather propose to create a new optional *interactorSet*

element in the *interactorList* of an *entry* element. In the *interactorSet* element, we would have :

- A required id attribute so we can refer to this element in the participant of an interaction (compact flavour).
- An optional *names* element. In the interactor element, the names element is required but it does not seem necessary for the interactorSet element.
- An optional *xref* element. It could contain external xrefs in case several databases refer to the same interactor set.
- A required *interactorSetType* element to describe the kind of interactor set (open set, ...). This element would be a *CvType* and could be described in the PSI-MI ontology.
- A required *interactorCandidateList* element which would contain one to many *interactorRef* elements. The *interactorRef* element would refer to the id of an interactor in the *interactorList* but not to any other *interactorSet* id in the *interactorList*.
- An optional *attributeList* element to add some properties/comments to the interactorSet
- Other optional properties such as features, stoichiometry?

Notes: *Describing the interactorSet in an interactorList only makes sense if several interactions may share the same interactorSet as a participant. It does not make any sense if we want to describe binding sites of each interactor candidate within the same interaction. We also need to know if we only want to refer to an interactor id or if we want to give more information such as a probability, etc.*

Example:

```
−<interactorList>
  +<interactor id="1035268"></interactor>
  +<interactor id="1035269"></interactor>
  +<interactor id="1035270"></interactor>
  +<interactor id="1035271"></interactor>
  +<interactor id="1035272"></interactor>
  +<interactor id="1035273"></interactor>
  −<interactorSet id="1">
    −<xref>
       <primaryRef refTypeAc="MI:0356" refType="identity" id="EBI-3018345555" dbAc="MI:0469" db="intact"/>
     </xref>
    −<interactorSetType>
      −<names>
         <shortLabel>open set</shortLabel>
         <fullName>open set</fullName>
       </names>
      −<xref>
         <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:xxxx" dbAc="MI:0488" db="psi-mi"/>
       </xref>
     </interactorSetType>
    −<interactorCandidateList>
       <interactorRef>1035270</interactorRef>
       <interactorRef>1035271</interactorRef>
       <interactorRef>1035272</interactorRef>
       <interactorRef>1035273</interactorRef>
     </interactorCandidateList>
   </interactorSet>
 </interactorList>
```

In the *participant* element, we could refer to the *interactorSet* id using the existing *interactorRef* element (so *interactorRef* would point to unique interactor id or *interactorSet* id in the *interactorList* element).

Example:

```
−<participantList>
  +<participant id="1035275"></participant>
  −<participant id="1035276">
    +<names></names>
    +<xref></xref>
      <interactorRef>1</interactorRef>
    +<biologicalRole></biologicalRole>
    +<experimentalRoleList></experimentalRoleList>
    </participant>
  </participantList>
```

To keep the same logic compact/expanded in the *participant* element, we could propose either an *interactorRef (compact flavour for interactor and interactorSet)* element, an *interactor (expanded flavour for a simple interactor)* element, or an *interactionRef (compact flavour for interaction)* element or an *interactorSet (expanded flavour for interactorSet)* element. If we choose to describe *interactorSet* in the *participant*, we need to describe each interactor candidate in the interactorSet because we don't have a list of interactors to refer to.

Example (interactorSet in a participant, expanded flavour):

```
−<participant id="1035276">
  +<names></names>
  +<xref></xref>
  −<interactorSet id="1">
    −<xref>
       <primaryRef refTypeAc="MI:0356" refType="identity" id="EBI-3018345555" dbAc="MI:0469" db="intact"/>
     </xref>
    −<interactorSetType>
      −<names>
         <shortLabel>open set</shortLabel>
         <fullName>open set</fullName>
       </names>
      −<xref>
         <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:xxxx" dbAc="MI:0488" db="psi-mi"/>
       </xref>
     </interactorSetType>
    −<interactorCandidateList>
      −<interactor id="1035270">
        +<names></names>
        +<xref></xref>
        +<interactorType></interactorType>
        +<organism ncbiTaxId="9606"></organism>
        +<sequence></sequence>
       </interactor>
      +<interactor id="1035271"></interactor>
      +<interactor id="1035272"></interactor>
      +<interactor id="1035273"></interactor>
     </interactorCandidateList>
   </interactorSet>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
</participant>
```

b.2) Proposal 2

We propose to create a new *interactorSet* element as an alternative to *interactorRef*, *interactor* and *interactionRef* in the *participant* element. In the *interactorSet* element, we would have :

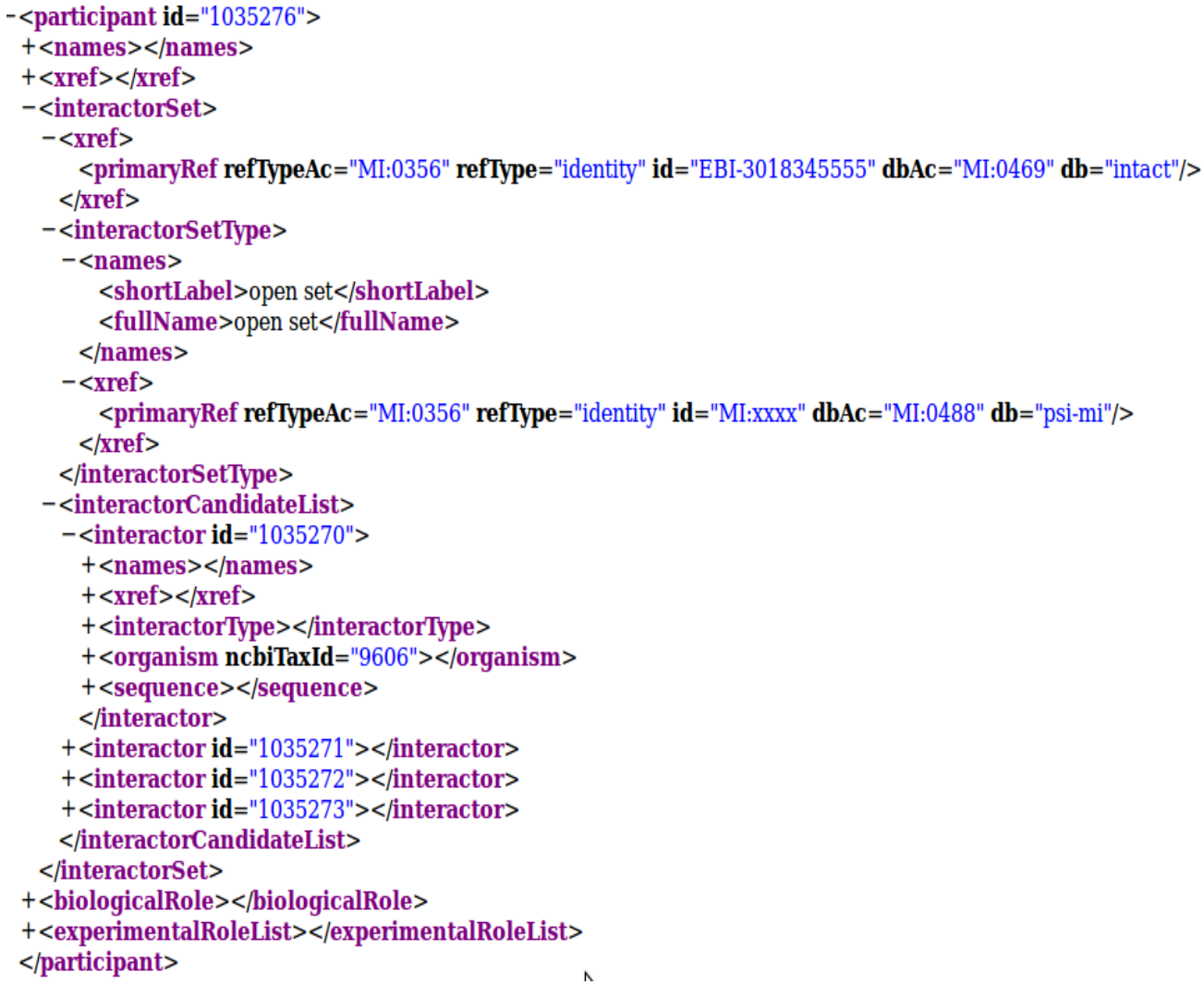
- An optional *names* element. In the interactor element, the names element is required but it does not seem necessary for the interactorSet element.
- An optional *xref* element. It could contain external xrefs in case several databases refer to the same interactor set.
- A required *interactorSetType* element to describe the kind of interactor set (open set, ...). This element would be a *CvType* and could be described in the PSI-MI ontology.
- A required *interactorCandidateList* element which would contain one to many *interactorRef* elements (compact flavour) or one to many *interactor* elements (expanded flavour).
- An optional *attributeList* element to add some properties/comments to the interactorSet.
- Other optional properties such as features, stoichiometry?

Notes: *Describing the interactorSet only in the participant is interesting when an interactorSet cannot be re-used for another participant in another interaction and needs. However, it would not be possible to describe different binding sites for each interactor candidate within the same participant.*

Example of interactorSet (compact flavour):

```
−<participant id="1035276">
  +<names></names>
  +<xref></xref>
  −<interactorSet>
    −<xref>
      <primaryRef refTypeAc="MI:0356" refType="identity" id="EBI-3018345555" dbAc="MI:0469" db="intact"/>
    </xref>
    −<interactorSetType>
      −<names>
        <shortLabel>open set</shortLabel>
        <fullName>open set</fullName>
      </names>
      −<xref>
        <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:xxxx" dbAc="MI:0488" db="psi-mi"/>
      </xref>
    </interactorSetType>
    −<interactorCandidateList>
      <interactorRef>1035270</interactorRef>
      <interactorRef>1035271</interactorRef>
      <interactorRef>1035272</interactorRef>
      <interactorRef>1035273</interactorRef>
    </interactorCandidateList>
  </interactorSet>
  +<biologicalRole></biologicalRole>
  +<experimentalRoleList></experimentalRoleList>
</participant>
```

Example of interactorSet (expanded flavour):



b.3) Proposal 3

We propose to create a new *participantSet* element as an alternative to *participant* in the *participantList* element. In the *participantSet* element, we would have :

- An optional *xref* element. It could contain external xrefs in case several databases refer to the same participant set.
- A required *participantSetType* element to describe the kind of interactor set (open set, ...). This element would be a *CvType* and could be described in the PSI-MI ontology.

- A required *participantCandidateList* element which would contain one to many *participant* elements.
- An optional *attributeList* element to add some properties/comments to the participantSet.
- Other participant properties that would also make sense for a participant set? biological role, experimental role, stoichiometry, etc?

Notes: *Describing the participantSet only in the participantList is interesting when we want to describe a set of interacting molecules that are candidates and each of them have different features/stoichiometry/etc.. However, it would not be possible to re-use the same participantSet in another interaction.*

Example of participantSet:

```
−<participantList>
  +<participant id="1035275"></participant>
  −<participantSet>
    −<xref>
       <primaryRef refTypeAc="MI:0356" refType="identity" id="EBI-6503450" dbAc="MI:0469" db="intact"/>
     </xref>
    −<participantSetType>
      −<names>
         <shortLabel>open set</shortLabel>
         <fullName>open set</fullName>
       </names>
      −<xref>
         <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:xxxx" dbAc="MI:0488" db="psi-mi"/>
       </xref>
     </participantSetType>
    −<participantCandidateList>
      +<participant id="1035275"></participant>
      +<participant id="1035276"></participant>
     </participantCandidateList>
    −<biologicalRole>
      −<names>
         <shortLabel>unspecified role</shortLabel>
         <fullName>unspecified role</fullName>
       </names>
      −<xref>
         <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:0499" dbAc="MI:0488" db="psi-mi"/>
         <secondaryRef refTypeAc="MI:0356" refType="identity" id="EBI-77781" dbAc="MI:0469" db="intact"/>
         <secondaryRef refTypeAc="MI:0358" refType="primary-reference" id="14755292" dbAc="MI:0446" db="pubmed"/>
       </xref>
     </biologicalRole>
   </participantSet>
</participantList>
```

9) Complexes

a) Context

The PSI-XML schema was designed to represent experimental interactions, therefore an experiment description is required for each interaction (MIMIx). Stable complexes are however not ‘experimental’ and rather than attaching them to fake experiments and fake publications with “inferred by curator” or “inferred by author” for the interaction detection method, we would prefer to refer to a list of interaction evidences in the *Xref* element and leave the complex without an experiment.
Sub-complexes can already be included into large complexes because participants can refer to interactions. However, the schema cannot describe complex binding sites when a participant of a subunit is binding to the participant of another subunit or when we have composite binding sites.

Some examples about how IntAct currently exports complexes in PSI-XML 2.5 can be found here.

b) Propositions

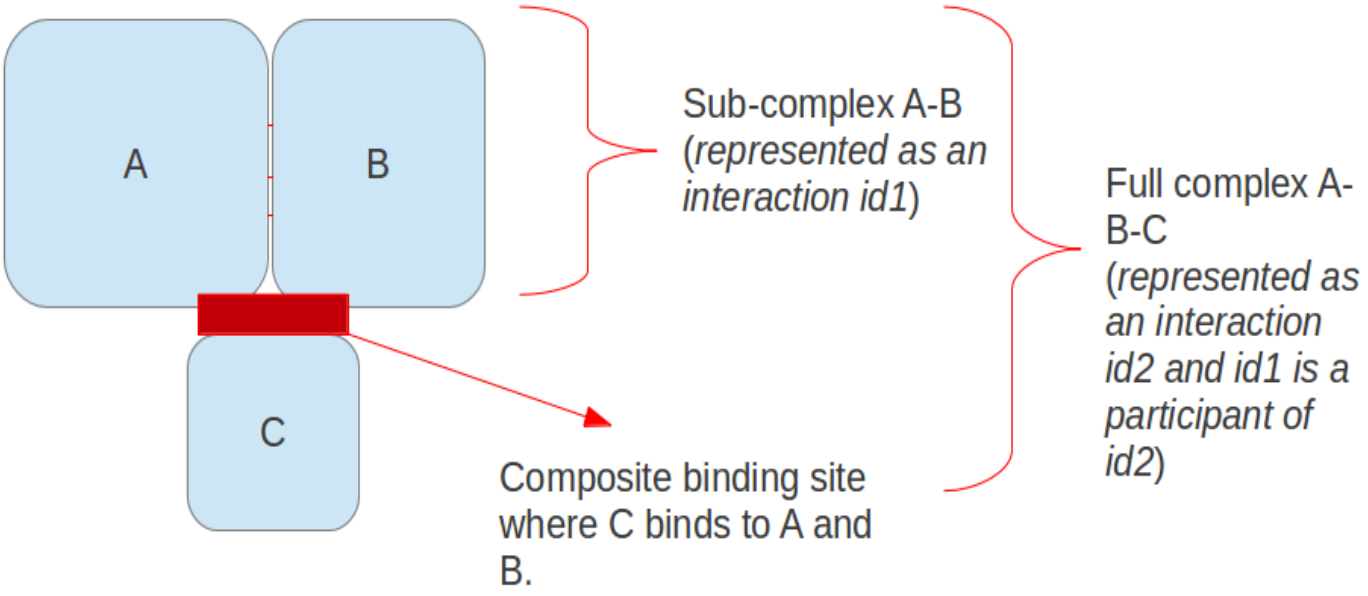
b.1) Complex binding sites

First, we would like to represent complex binding sites such as composite binding sites where a feature can refer to a participant of a sub-complex. We propose to add an optional *participantRef* element in the *featureRange* element so each *featureRange* of a *feature* can points to a specific *participant* in the *entry*.

The updated XSD is available here.

For instance, we have a sub-complex A-B that needs to assembly before binding to C.

Example: composite binding site



In the *interaction* id2, the *participant* C will have a *feature* binding site F1 and the *participant* A-B will also have a *feature* binding site F2. In the *interaction* id2, we will have an *inferredInteraction* element which will link F1 and F2. However, to be able to say that F2 is composed of a sequence portion from A and another sequence portion from B, we need to add a *participantRef* element for each *featureRange* element of F2. F2 will have two *featureRanges* elements R1 and R2. R1 will have a *participantRef* to A in sub-complex A-B and R2 will have a *participantRef* to B in complex A-B.

Example: XML representation of sub-complex A-B

```
−<interaction id="1">
  −<names>
     <shortLabel>Complex A-B</shortLabel>
   </names>
  +<xref></xref>
  +<experimentList></experimentList>
  −<participantList>
    −<participant id="3">
      −<names>
         <shortLabel>Participant A</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180654</interactorRef>
      +<biologicalRole></biologicalRole>
      +<experimentalRoleList></experimentalRoleList>
      </participant>
    −<participant id="4">
      −<names>
         <shortLabel>Participant B</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180655</interactorRef>
      +<biologicalRole></biologicalRole>
      +<experimentalRoleList></experimentalRoleList>
      </participant>
   </participantList>
  +<interactionType></interactionType>
   <modelled>false</modelled>
   <intraMolecular>false</intraMolecular>
   <negative>false</negative>
  +<attributeList></attributeList>
  </interaction>
```

Example: XML representation of complex A-B-C

Example: XML representation of feature F2

```
−<feature id="7">
  −<names>
     <shortLabel>Feature F2</shortLabel>
   </names>
  +<xref></xref>
  +<featureType></featureType>
  −<featureRangeList>
    −<featureRange>
      +<startStatus></startStatus>
       <begin position="33"/>
      +<endStatus></endStatus>
       <end position="36"/>
       <isLink>false</isLink>
       <participantRef>3</participantRef>
     </featureRange>
    −<featureRange>
      +<startStatus></startStatus>
       <begin position="58"/>
      +<endStatus></endStatus>
       <end position="72"/>
       <isLink>false</isLink>
       <participantRef>4</participantRef>
     </featureRange>
   </featureRangeList>
</feature>
```

b.2) Complexes as an 'abstract' interaction

When representing complexes, there are some information that are not wanted because we want to describe an 'abstract' interaction which can then refer to several interaction evidences from several external interaction databases.

- *Experiment with a fake interaction detection method and a fake publication.*
- *Availability does not make sense, it is not an experimental interaction*
- *Negative does not make sense for complexes*
- *Modelled is not necessary and does not bring any information to*

complexes

- *Participant identification method does not make sense (always predetermined)*
- *Participant experimental roles do not make sense as we are representing an 'abstract' interaction*
- *Participant experimental preparations are only for interaction evidences*
- *Participant experimental interactor does not make sense for complexes*
- *Participant host organisms does not make sense as the participant refers to a biological molecule that has not been expressed by an experimental cell line.*
- *Is a participant confidence list useful for describing complexes?*
- *Is a participant parameter list useful for describing complexes?*
- *Feature detection method does not make sense as we are representing an 'abstract' interaction.*
- *Feature experiment ref list.*

The updated XSD is available here.

b.2.1) Proposal 1

Among the elements described above, only the *experimentList* is mandatory in the *interaction* element. We could make *experimentList* optional and then when we describe complexes, we simply make sure that none of the elements listed above are provided. Only external cross references to existing interaction evidences could be provided in the *Xref* element to point to experimental evidences.

We would also need to add an optional *organism* element to describe the organism of the complex.

We would also need to add an optional evidenceType element to describe the ECO code for instance as Cv term

We would also need to add an optional *interactorType* element to

describe the interactor type when this complex is used as an interactor.

Notes: *This solution is not backward compatible with the PSI-XML 2.5 schema and is very messy as we don't have a clear way to distinguish complexes from interaction evidences (only with interaction type?). It may leads to a lot of inconsistencies if users export interaction evidences without experiment but with a participant identification method, etc. In addition to that, organism and interactorType elements only make sense in the case of an abstract interaction or complex where we want to use it as an interactor. All the consistency would have to be implemented as semantic rules in the PSI-MI validator to make sure that the exported interactions make sense which would not be easy as there no simple way to distinguish complexes from interaction evidences.*

Example:

One complex A-B without experiment. One Xref to the interaction evidence in IntAct with qualifier 'exp-evidence'. Participants without experimental roles.

```
−<interaction id="1">
  −<names>
     <shortLabel>Complex A-B</shortLabel>
   </names>
  +<xref></xref>
  −<participantList>
    −<participant id="3">
      −<names>
         <shortLabel>Participant A</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180654</interactorRef>
      +<biologicalRole></biologicalRole>
      </participant>
    −<participant id="4">
      −<names>
         <shortLabel>Participant B</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180655</interactorRef>
      +<biologicalRole></biologicalRole>
      </participant>
   </participantList>
  −<organism ncbiTaxId="9606">
    +<names></names>
   </organism>
  +<interactionType></interactionType>
  −<interactorType>
    −<names>
       <shortLabel>protein complex</shortLabel>
       <fullName>protein complex</fullName>
     </names>
    +<xref></xref>
   </interactorType>
   <modelled>false</modelled>
   <intraMolecular>false</intraMolecular>
   <negative>false</negative>
  +<attributeList></attributeList>
 </interaction>
```

b.2.2) Proposal 2

We propose to add a new optional *modelledInteraction* element in the *interactionList.* This element would be used to describe ‘abstract’ or ‘modelled’ interactions such as stable complexes, allosteric interactions, etc.

The *modelledInteraction* element would contain :

- Required *id* attribute that has to be unique in the all *entry*.
- Required *Names* element for describing complex names and synonyms.
- Optional *Xref* element to point to external interaction evidences, pathway databases, GO ontology, etc.
- Optional *bindingFeatureList* to describe binding sites and topology
- One Optional *interactionType* element to describe the kind of complex or cooperative/enzymatic reaction.
- One optional interactorType element to describe what kind of interactor the complex is. For instance, we want to distinguish protein complex from protein-rna complex, etc.
- Optional *intramolecular* element.
- Optional *confidenceList* element to describe confidences in this complex. Each *confidence* element would not have a *experimentRefList* but an optional *bibRef* or *Xref* element to points to the original publication.
- Optional *parameterList* element to describe parameters in this complex. Each *parameter* element would not have a *experimentRefList* but an optional *bibRef* or *Xref* element to points to the original publication.
- Optional *attributeList* to add comments and other information about the complex.
- Optional *evidenceType* node to describe the ECO code of this complex (physical evidence, inferred from literature, etc.)
- Required *organism* element to describe the organism of the complex.
- Required *participantList* which will describe the *participants* of a

modelledInteraction.

The *participant* element of a *modelledInteraction* would contain :

- Required *id* attribute that has to be unique in the all *entry*.
- An optional *Xref* element. The Names element would not be useful so it would not be allowed?
- A required choice between *interactorRef*, *interactor* and *modelledInteractionRef* element if we want to refer to a sub-complex.
- An optional *biologicalRole* element to describe possible biological role of a participant in a complex.
- An optional *attributeList* element to add more information about a specific participant in a complex.
- An optional *featureList* where we want to describe binding sites and PTM.
- Depending on what has been decided with participant stoichiometry, either optional stoichiometry, minStoichiometry and maxStoichiometry attributes or an optional stoichiometry and stoichiometryRange element (see section 3).

The *feature* element of a participant *in a modelledInteraction* would contain :

- An optional *Names* element.
- An optional *Xref* element to points to specific domains databases such as interpro.
- An optional *featureType* element (or required?) or *featureTypeList* depending on what is decided about interaction dependencies such a prerequisite PTM, etc (see section 6).
- An optional *attributeList* element
- A required *featureRangeList* element to describe ranges. The range elements would be the same as in a normal *interaction* element.

Notes: *This solution is good to be able to quickly distinguish a 'modelledInteraction' such as a complex from an interaction evidence. It also makes sure that a user will not mix experimental details with the description of an abstract interaction such as a complex. However, it may be confusing to have two elements 'interaction' and 'modelledInteraction' in the same interactionList.*

Example: two modelledInteraction in the interactionList

```
-<interactionList>
  +<interaction id="180659"></interaction>
  +<interaction id="180669"></interaction>
  +<interaction id="180673"></interaction>
  +<interaction id="180678"></interaction>
  +<interaction id="180682"></interaction>
  +<interaction id="180688"></interaction>
  +<modelledInteraction id="1"></modelledInteraction>
  +<modelledInteraction id="2"></modelledInteraction>
 </interactionList>
```

Example: use modelledInteraction to represent complex A-B

```
−<modelledInteraction id="1">
  −<names>
     <shortLabel>Complex A-B</shortLabel>
   </names>
  −<xref>
     <primaryRef refType="exp-evidence" id="EBI-475939" dbAc="MI:0469" db="intact"/>
   </xref>
  −<participantList>
    −<participant id="3">
      −<names>
         <shortLabel>Participant A</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180654</interactorRef>
      +<biologicalRole></biologicalRole>
      </participant>
    −<participant id="4">
      −<names>
         <shortLabel>Participant B</shortLabel>
       </names>
      +<xref></xref>
       <interactorRef>180655</interactorRef>
      +<biologicalRole></biologicalRole>
      </participant>
   </participantList>
  +<organism ncbiTaxId="9606"></organism>
  +<interactionType></interactionType>
  −<interactorType>
    −<names>
       <shortLabel>protein complex</shortLabel>
       <fullName>protein complex</fullName>
     </names>
    +<xref></xref>
   </interactorType>
   <intraMolecular>false</intraMolecular>
  +<confidenceList></confidenceList>
  +<parameterList></parameterList>
  +<attributeList></attributeList>
 </modelledInteraction>
```

Example: use confidenceList and paramaterList in

modelledInteraction

```
−<confidenceList>
  −<confidence>
    −<type>
      −<names>
          <shortLabel>complex score</shortLabel>
        </names>
      −<xref>
          <primaryRef refTypeAc="MI:0356" refType="identity" id="MI:xxxx" dbAc="MI:0488" db="psi-mi"/>
        </xref>
      </type>
      <value>0.66</value>
    </confidence>
  </confidenceList>
−<parameterList>
    <parameter factor="2.34" exponent="-5" base="10" unitAc="MI:0648" unit="molar" termAc="MI:0646" term="kd"> </parameter>
  </parameterList>
```

Example: use modelledInteraction to represent complex A-B-C

```
−<modelledInteraction id="2">
  −<names>
     <shortLabel>Complex A-B-C</shortLabel>
   </names>
  −<participantList>
    −<participant id="5">
      −<names>
         <shortLabel>Participant A-B (sub-complex)</shortLabel>
       </names>
       <modelledInteractionRef>1</modelledInteractionRef>
      +<biologicalRole></biologicalRole>
      −<featureList>
       +<feature id="7"></feature>
       </featureList>
      </participant>
    −<participant id="6">
      −<names>
         <shortLabel>Participant C</shortLabel>
       </names>
       <interactorRef>180655</interactorRef>
      +<biologicalRole></biologicalRole>
      −<featureList>
       +<feature id="8"></feature>
       </featureList>
      </participant>
   </participantList>
  +<organism ncbiTaxId="9606"></organism>
  −<bindingFeatureList>
    −<bindingFeatures>
       <participantFeatureRef>7</participantFeatureRef>
       <participantFeatureRef>8</participantFeatureRef>
     </bindingFeatures>
   </bindingFeatureList>
  +<interactionType></interactionType>
  +<interactorType></interactorType>
   <intraMolecular>false</intraMolecular>
 </modelledInteraction>
```

10) Cooperative interactions

a) Context

We would like to represent cooperative interactions that are a set of molecular binding events that influence each other either positively or negatively through allostery or pre-assembly. In this context, covalent post-translational modifications are considered as binding events. Describing cooperative interactions is difficult as it can be a mixture of both molecular interaction and pathway/process.

Currently, cooperativity is only captured using annotations at the interaction level which is not great for parsing such a complex process. We also have the same issues as for the stable complexes: we don't describe experimental interactions so the experimental details such as interaction detection method are not relevant.

Example: preassembly

```
−<interaction id="11">
  −<names>
      <shortLabel>Dp1-E2F1</shortLabel>
    </names>
  +<xref></xref>
  −<experimentList>
      <experimentRef>41</experimentRef>
    </experimentList>
  +<participantList></participantList>
  −<interactionType>
    −<names>
        <shortLabel>direct interaction</shortLabel>
      </names>
    +<xref></xref>
    </interactionType>
  −<attributeList>
      <attribute name="pre-assembly" nameAc="MI:1158"/>
      <attribute name="affected interaction" nameAc="MI:1150">12</attribute>
      <attribute name="positive cooperative effect" nameAc="MI:1154"/>
      <attribute name="composite binding site formation" nameAc="MI:1171"/>
    </attributeList>
  </interaction>
−<interaction id="12">
  −<names>
      <shortLabel>Dp1_E2F1-RbCc</shortLabel>
    </names>
  +<xref></xref>
  +<experimentList></experimentList>
  +<participantList></participantList>
  +<interactionType></interactionType>
  −<attributeList>
      <attribute name="pre-assembly" nameAc="MI:1158"/>
      <attribute name="affected interaction" nameAc="MI:1150">13</attribute>
      <attribute name="affected interaction" nameAc="MI:1150">14</attribute>
      <attribute name="positive cooperative effect" nameAc="MI:1154"/>
      <attribute name="configurational pre-organization" nameAc="MI:1174"/>
    </attributeList>
  </interaction>
```

Example: allostery

```
−<interaction id="18">
  −<names>
      <shortLabel>RbN-RbP</shortLabel>
    </names>
  +<xref></xref>
  +<experimentList></experimentList>
  +<participantList></participantList>
  +<inferredInteractionList></inferredInteractionList>
  +<interactionType></interactionType>
    <intraMolecular>true</intraMolecular>
  −<attributeList>
      <attribute name="allostery" nameAc="MI:1157"/>
      <attribute name="affected interaction" nameAc="MI:1150">14</attribute>
      <attribute name="negative cooperative effect" nameAc="MI:1155"/>
      <attribute name="allosteric molecule" nameAc="MI:1159">36</attribute>
      <attribute name="allosteric effector" nameAc="MI:1159">35</attribute>
      <attribute name="allosteric k-type response" nameAc="MI:1162"/>
      <attribute name="heterotropic allostery" nameAc="MI:1168"/>
    </attributeList>
  </interaction>
```

b) Propositions

We propose to add an optional *cooperativeEffectList* in the *interaction* element (if we decide to go with proposal 1 for representing complexes) or the *modelledInteraction* element (if we decide to go with proposal 2 for representing complexes).

In the *cooperativeEffectList* we can either use an *allostery* element to describe allosteric interactions or a *preassembly* element to describe other cooperative effects.

Both *preassembly* and *allostery* elements would contain :

- A required *cooperativityEvidenceList* element that would contain one to many *cooperativityEvidenceDescription* element(s).
- One required *affectedInteractionList* element which would contain one to many *affectedInteraction* element(s). Each *affectedInteraction*

element would refer to the id of the model interaction that is affected by the current model interaction.

- One required *cooperativeEffectOutcome* element to indicate whether the cooperative effect is positive (either induced or enhanced) or negative (either inhibited or abrogated). This element is controlled by the PSI-MI controlled vocabulary "cooperative effect outcome" root term, id MI:1153.
- One optional *cooperativeEffectResponse* element to indicate how the cooperative mechanism affects another interaction. This element is controlled by the PSI-MI controlled vocabulary "cooperative effect response" root term.
- One optional *cooperativeEffectValue* element to describe a decimal quantification of the cooperative effect. The ratio of a binding or a catalytic parameter of the affected interaction in the absence versus presence of the affecting interaction.
- One optional *attributeList* element in case we want to add more information to the cooperative effect

The *cooperativityEvidenceDescription* element would contain :

- A required *bibRef* element to describe the publication where the cooperative effect has been shown.
- An optional *evidenceMethodList* that would contain one to many *evidenceMethod* element(s) which describe the experimental methods from which this cooperative effect has been inferred.

Example: preassembly (related to previous example, interaction id 11)

```
−<cooperativeEffectList>
  −<preassembly>
    −<cooperativityEvidenceList>
      −<cooperativityEvidenceDescription>
        −<bibref>
          −<xref>
            <primaryRef refType="primary-reference" db="pubmed" refTypeAc="MI:0358" dbAc="MI:0446" id="16360038"/>
          </xref>
        </bibref>
      </cooperativityEvidenceDescription>
    </cooperativityEvidenceList>
    −<affectedInteractionList>
      <affectedInteraction>12</affectedInteraction>
    </affectedInteractionList>
    −<cooperativeEffectOutcome>
      −<names>
        <shortLabel>positive cooperative effect</shortLabel>
      </names>
      −<xref>
        <primaryRef db="psi-mi" dbAc="MI:0488" id="MI:1154" refType="identity" refTypeAc="MI:0356"/>
      </xref>
    </cooperativeEffectOutcome>
    −<cooperativeEffectResponse>
      −<names>
        <shortLabel>composite binding site formation</shortLabel>
      </names>
      −<xref>
        <primaryRef db="psi-mi" dbAc="MI:0488" id="MI:1171" refType="identity" refTypeAc="MI:0356"/>
      </xref>
    </cooperativeEffectResponse>
  </preassembly>
</cooperativeEffectList>
```

The *allostery* element would be an extension of the *preassembly* element and would contain more specific elements :

- One required *allostericMolecule* element which refers to the id of the participant that is allosterically regulated.
- One required choice between *allostericEffector* (refers to the id of the participant that elicits an allosteric response in an allosteric molecule upon binding to that molecule) and *allostericModification* (refers to id of the feature that elicits an allosteric response) elements.
- One optional *allostericMechanism* element to indicate the type of changes that occur in an allosteric molecule upon allosteric modification or binding of an allosteric effector and result in an allosteric response. This element is controlled by the PSI-MI

controlled vocabulary "allosteric mechanism", root term id MI:1164.

- One optional *allosteryType* element to indicate the chemical relationship between the ligands whose binding is allosterically coupled. This element is controlled by the PSI-MI controlled vocabulary "allostery type", root term id MI:1167.

Example: allostery (related to previous example, interaction id 18)

```
<cooperativeEffectList>
 <allostery>
  <cooperativityEvidenceList>
   <cooperativityEvidenceDescription>
    <bibref>
     <xref>
      <primaryRef refType="primary-reference" db="pubmed" refTypeAc="MI:0358" dbAc="MI:0446" id="22569856"/>
     </xref>
    </bibref>
   </cooperativityEvidenceDescription>
  </cooperativityEvidenceList>
  <affectedInteractionList>
   <affectedInteraction>14</affectedInteraction>
  </affectedInteractionList>
  <allostericMolecule>36</allostericMolecule>
  <allostericEffector>35</allostericEffector>
  <cooperativeEffectOutcome>
   <names>
    <shortLabel>negative cooperative effect</shortLabel>
   </names>
   <xref></xref>
  </cooperativeEffectOutcome>
  <cooperativeEffectResponse>
   <names>
    <shortLabel>allosteric k-type response</shortLabel>
   </names>
   <xref></xref>
  </cooperativeEffectResponse>
  <allosteryType>
   <names>
    <shortLabel>heterotropic allostery</shortLabel>
   </names>
   <xref>
    <primaryRef db="psi-mi" dbAc="MI:0488" id="MI:1168" refType="identity" refTypeAc="MI:0356"/>
   </xref>
  </allosteryType>
 </allostery>
</cooperativeEffectList>
```

The updated XSD is available here.

11) Capturing dynamic interactions in an experiment

a) Context

We want to represent dynamic interactions in the context of one to several experiments. Some interactions may occur in specific conditions defined by the experiment.

IntAct currently captures this information using annotations with topic ‘variable’ and ‘variable2’. The annotations at the level of the experiment are used to describe the variable conditions and the annotations at the level of the interaction describe the value of this conditions for a specific interaction. Of course this model is not optimal as it can only contain 2 variable conditions max, we cannot give an order to the eventsand we cannot represent negative results (no interactions observed for a specific set of conditions).

Example: experiment with one variable condition (cell cycle)

```
−<experimentDescription id="488920">
  +<names></names>
  +<bibref></bibref>
  +<xref></xref>
  +<hostOrganismList></hostOrganismList>
  +<interactionDetectionMethod></interactionDetectionMethod>
  +<participantIdentificationMethod></participantIdentificationMethod>
  −<attributeList>
    +<attribute nameAc="MI:0636" name="author-list"></attribute>
    +<attribute nameAc="MI:0627" name="exp-modification"></attribute>
      <attribute nameAc="MI:0634" name="contact-email">james_decaprio@dfci.havard.edu</attribute>
      <attribute name="variable">Cell cycle</attribute>
      <attribute nameAc="MI:0885" name="journal">Mol. Cell (1097-2765)</attribute>
      <attribute nameAc="MI:0886" name="publication year">2007</attribute>
    </attributeList>
</experimentDescription>
```

Example: interactions with one variable condition

```
−<interaction id="489127">
  +<names></names>
  +<xref></xref>
  −<experimentList>
      <experimentRef>488920</experimentRef>
    </experimentList>
  +<participantList></participantList>
  +<interactionType></interactionType>
    <modelled>false</modelled>
    <intraMolecular>false</intraMolecular>
    <negative>false</negative>
  −<attributeList>
      <attribute nameAc="MI:0599" name="figure legend">Table 1</attribute>
      <attribute name="variable_condition">G0 phase</attribute>
    </attributeList>
  </interaction>
−<interaction id="489142">
  +<names></names>
  +<xref></xref>
  −<experimentList>
      <experimentRef>488920</experimentRef>
    </experimentList>
  +<participantList></participantList>
  +<interactionType></interactionType>
    <modelled>false</modelled>
    <intraMolecular>false</intraMolecular>
    <negative>false</negative>
  −<attributeList>
      <attribute nameAc="MI:0599" name="figure legend">Fig. 2 & Table S3</attribute>
      <attribute name="variable_condition">S phase</attribute>
    </attributeList>
  </interaction>
```

b) Propositions

In the experiment element, we propose to add an optional variableParameterList element which would contain one to many variableParameter elements.

Each variableParameter element would contain:

- Required *description* element to describe the variable condition.
- Optional *unit* element to describe the unit of the different parameters in *variableValueList.*
- Required *variableValueList* element to list all the existing variable parameter values used in the experiment and we can specify an order. It would contain one to many *variableValue* elements.

  Each *variableValue* element would contain:

- Required *id* attribute that has to be unique in the all entry. It would be used to refer to the *variableValue* in the *interaction*.
- Optional *order* attribute to give an explicit order of the *variableValue* in the *variableValueList.*
- Required *value* element to describe the variable parameter value.

Example: experiment with one variable condition (cell cycle) and two different conditionValues

```
−<experimentDescription id="488920">
  +<names></names>
  +<bibref></bibref>
  +<xref></xref>
  +<hostOrganismList></hostOrganismList>
  +<interactionDetectionMethod></interactionDetectionMethod>
  +<participantIdentificationMethod></participantIdentificationMethod>
  −<variableParameterList>
    −<variableParameter>
      <description>Cell cycle</description>
      −<variableValueList>
        <variableValue id="1" order="1">GO phase</variableValue>
        <variableValue id="2" order="2">S phase</variableValue>
      </variableValueList>
    </variableParameter>
  </variableParameterList>
  −<attributeList>
    +<attribute nameAc="MI:0636" name="author-list"></attribute>
    +<attribute nameAc="MI:0627" name="exp-modification"></attribute>
    <attribute nameAc="MI:0634" name="contact-email">james_decaprio@dfci.havard.edu</attribute>
    <attribute nameAc="MI:0885" name="journal">Mol. Cell (1097-2765)</attribute>
    <attribute nameAc="MI:0886" name="publication year">2007</attribute>
  </attributeList>
</experimentDescription>
```

In the *interaction* element, we would add a new optional *experimentalVariableValueList* element that would contain one to many *variableValueRef* elements. Each of them would refer to the id of a *variableValue* element described in the experiment(s) attached to the interaction.

Example:

```
−<interaction id="489127">
  +<names></names>
  +<xref></xref>
  −<experimentList>
      <experimentRef>488920</experimentRef>
    </experimentList>
  −<experimentalVariableValueList>
    −<variableValues>
        <variableValueRef>1</variableValueRef>
    </variableValues>
  </experimentalVariableValueList>
  +<participantList></participantList>
  +<interactionType></interactionType>
    <modelled>false</modelled>
    <intraMolecular>false</intraMolecular>
    <negative>false</negative>
  +<attributeList></attributeList>
  </interaction>
−<interaction id="489142">
  +<names></names>
  +<xref></xref>
  −<experimentList>
      <experimentRef>488920</experimentRef>
    </experimentList>
  −<experimentalVariableValueList>
    −<variableValues>
        <variableValueRef>2</variableValueRef>
    </variableValues>
  </experimentalVariableValueList>
  +<participantList></participantList>
  +<interactionType></interactionType>
    <modelled>false</modelled>
    <intraMolecular>false</intraMolecular>
    <negative>false</negative>
  +<attributeList></attributeList>
  </interaction>
```

The XSD changes are available here.

12) Capturing causal relationships

a) Context

We want to represent causal relationships within a specific interaction. Ex: ***P48084 is required for phosphorylation of P38074 by P22204***

IntAct currently captures this information using attributes with specific topic ‘causality statement’ and the attribute text is formatted as described here.

Of course this model is not optimal as attributes are normally for free text and we may want a better/more structured representation. It may also be difficult to keep this information up to date with the uniprot update where uniprot ids can become secondary ids, etc.

NOTE: We may want to unify the way we represent cooperative effect and causal relationships as we may have some overlap.

Example:

b) Propositions

All the causal relationships are in the context of an interaction parent. The structure of a causal relationship would be :

- **source participant**: the participating molecule/complex/interactorSet which is the source of the causal relationship
- **causality statement**: will be a controlled vocabulary term which describes the causality statement (Ex: is required for, increases activity of, etc.)
- **target participant** : the participating molecule/complex/interactorSet which is the target of the causal relationship

The possible XSD changes are available here.

b.1) Proposal 1

All the causal relationships would be described in a *causalRelationshipList* element within the *interaction* element. The *causalRelationshipList* element would contain one to many *causalRelationship* element(s) which would be composed of :
- a *participantSourceRef* element which points to a participant id described in the interaction
- a *causalityStatement* element which would be a CvType element.
- a *participantTargetRef* element which points to a participant id described in the interaction

Example:

```
−<interaction id="979031">
  −<names>
     <shortLabel>erg-klk2</shortLabel>
   </names>
  +<xref></xref>
  +<experimentList></experimentList>
  −<participantList>
   +<participant id="979032"></participant>
   +<participant id="979033"></participant>
   </participantList>
  +<interactionType></interactionType>
   <modelled>false</modelled>
   <intraMolecular>false</intraMolecular>
   <negative>false</negative>
  −<causalRelationshipList>
   −<causalRelationship>
      <participantSourceRef>979033</participantSourceRef>
     −<causalityStatement>
      −<names>
         <shortLabel>inhibits expression at RNA level of</shortLabel>
       </names>
      −<xref>
         <primaryRef db="psi-mi" dbAc="MI:0488" primaryId="MI:xxxx" refType="identity" refTypeAc="MI:0356"/>
       </xref>
      </causalityStatement>
      <participantTargetRef>979032</participantTargetRef>
    </causalRelationship>
   </causalRelationshipList>
  +<attributeList></attributeList>
  </interaction>
```

Notes: *Describing the causal relationships at the interaction level may be interesting if we want to have a summary of the causal relationships per interaction. It would be the same logic as the inferredInteractionList in the interaction node.*

b.2) Proposal 2

All the causal relationships would be described in a *causalRelationshipList* element within the *interaction* element. The *causalRelationshipList* element would contain one to many *causalRelationship* element(s) which would be composed of :
- a *participantSourceRef* attribute which points to a participant id described in the interaction
- a *causalityStatement* attribute which would point to a controlledvocabulary name in the PSI-MI ontology.
- a *causalityStatementAc* attribute which would point to a valid MI identifier in the PSI-MI ontology.

- a *participantTargetRef* attribute which points to a participant id described in the interaction

Example:

```
−<interaction id="979031">
  −<names>
     <shortLabel>erg-klk2</shortLabel>
   </names>
  +<xref></xref>
  +<experimentList></experimentList>
  +<participantList></participantList>
  +<interactionType></interactionType>
   <modelled>false</modelled>
   <intraMolecular>false</intraMolecular>
   <negative>false</negative>
  −<causalRelationshipList>
     <causalRelationship participantSourceRef="979033" causalityStatement="inhibits expression at RNA level of" participantTargetRef="979032"/>
   </causalRelationshipList>
  −<attributeList>
     <attribute nameAc="MI:0599" name="figure legend">Fig 5A.</attribute>
    −<attribute nameAc="MI:0612" name="comment">
       AR and ERG cobinding to regulatory regions of representative AR target genes.
     </attribute>
   </attributeList>
 </interaction>
```

Notes: *Describing the causal relationships at the interaction level may be interesting if we want to have a summary of the causal relationships per interaction. It would be the same logic as the inferredInteractionList in the interaction node. Adding the information as attributes of the causalRelationship node make the XML more compact but it also makes it less flexible as we impose to use MI terms to describe the causality statement.*

b.3) Proposal 3

All the causal relationships would be described in a *causalRelationshipList* element within the *participant* element which would be the source of all the causality statements. The *causalRelationshipList* element would contain one to many *causalRelationship* element(s) which would be composed of :

- a *causalityStatement* element which would be a CvType element.
- a *participantTargetRef* element which points to a participant id described in the interaction

Example:

```
−<participantList>
  −<participant id="979032">
    +<names></names>
    +<xref></xref>
      <interactorRef>978959</interactorRef>
    +<biologicalRole></biologicalRole>
    +<experimentalRoleList></experimentalRoleList>
    </participant>
  −<participant id="979033">
    +<names></names>
    +<xref></xref>
      <interactorRef>978970</interactorRef>
    +<biologicalRole></biologicalRole>
    +<experimentalRoleList></experimentalRoleList>
    −<causalRelationshipList>
      −<causalRelationship>
        −<causalityStatement>
          −<names>
              <shortLabel>inhibits expression at RNA level of</shortLabel>
            </names>
          −<xref>
              <primaryRef db="psi-mi" dbAc="MI:0488" primaryId="MI:xxxx" refType="identity" refTypeAc="MI:0356"/>
            </xref>
          </causalityStatement>
          <participantTargetRef>979032</participantTargetRef>
        </causalRelationship>
      </causalRelationshipList>
    </participant>
  </participantList>
```

Notes: *Describing the causal relationships at the participant level may be interesting if we want to have a summary of the causal relationships per source participant (avoid another reference to resolve when parsing the XML file).*

b.4) Proposal 4

All the causal relationships would be described in a *causalRelationshipList* element within the *participant* element which would be the source of all the causality statements. The *causalRelationshipList* element would contain one to many *causalRelationship* element(s) which would be composed of :

- a *causalityStatement* attribute which would point to a controlledvocabulary name in the PSI-MI ontology.
- a *causalityStatementAc* attribute which would point to a valid MI identifier in the PSI-MI ontology.
- a *participantTargetRef* attribute which points to a participant id described

in the interaction

Example:

```
−<interaction id="979031">
  −<names>
     <shortLabel>erg-klk2</shortLabel>
   </names>
  +<xref></xref>
  +<experimentList></experimentList>
  −<participantList>
    −<participant id="979032">
      +<names></names>
      +<xref></xref>
        <interactorRef>978959</interactorRef>
      +<biologicalRole></biologicalRole>
      +<experimentalRoleList></experimentalRoleList>
      </participant>
    −<participant id="979033">
      +<names></names>
      +<xref></xref>
        <interactorRef>978970</interactorRef>
      +<biologicalRole></biologicalRole>
      +<experimentalRoleList></experimentalRoleList>
      −<causalRelationshipList>
          <causalRelationship causalityStatement="inhibits expression at RNA level of" participantTargetRef="979032"/>
        </causalRelationshipList>
      </participant>
    </participantList>
  +<interactionType></interactionType>
    <modelled>false</modelled>
    <intraMolecular>false</intraMolecular>
    <negative>false</negative>
  +<attributeList></attributeList>
  </interaction>
```

Notes: *Describing the causal relationships at the participant level may be interesting if we want to have a summary of the causal relationships per source participant (avoid another reference to resolve when parsing the XML file). Adding the information as attributes of the causalRelationship node make the XML more compact but it also makes it less flexible as we impose to use MI terms to describe the causality statement.*

XSD Annexes

a.1) *BibRef element*

Current XSD bibRef type:

```
<xs:complexType name="bibref">
    <xs:annotation>
        <xs:documentation>Bibliographic reference.</xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:element name="xref" type="mif:xref">
            <xs:annotation>
                <xs:documentation>Bibliographic reference in external database, usually PubMed.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList">
            <xs:annotation>
                <xs:documentation>Alternative description of bibliographic reference if no external database entry is available.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:choice>
</xs:complexType>
```

Updated XSD bibRef type:

```
<xs:complexType name="bibref">
    <xs:annotation>
        <xs:documentation>Bibliographic reference.</xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:sequence>
            <xs:element name="xref" type="mif:xref" minOccurs="1" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>Bibliographic reference in external database, usually PubMed.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="attributeList" type="mif:attributeList" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>Additional description of bibliographic reference such as publication title, authors, journal, publication date...</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
        <xs:sequence>
            <xs:element name="attributeList" type="mif:attributeList" minOccurs="1" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>Alternative description of bibliographic reference if no external database entry is available.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:choice>
</xs:complexType>
```

a.2) ExperimentRef in parameter

Current XSD parameter element:

```
<xs:complexType name="parameter">
    <xs:complexContent>
        <xs:extension base="mif:parameterBase">
            <xs:sequence>
                <xs:element name="experimentRef" type="xs:int">
                    <xs:annotation>
                        <xs:documentation>Reference to the experiment in which this parameter has been
                            determined. If not given, it is assumed that this is valid for all
                            experiments attached to the interaction.
                        </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="uncertainty" type="xs:decimal" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

Updated XSD parameter element:

```
<xs:complexType name="parameter">
    <xs:complexContent>
        <xs:extension base="mif:parameterBase">
            <xs:sequence>
                <xs:element name="experimentRef" type="xs:int" minOccurs="0">
                    <xs:annotation>
                        <xs:documentation>Reference to the experiment in which this parameter has been
                            determined. If not given, it is assumed that this is valid for all
                            experiments attached to the interaction.
                        </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="uncertainty" type="xs:decimal" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

a.3) Range position and interval type

Current XSD position type:

```
<xs:complexType name="position">
    <xs:attribute name="position" type="xs:unsignedLong" use="required"/>
</xs:complexType>
```

Current XSD interval type:

```
<xs:complexType name="interval">
    <xs:annotation>
        <xs:documentation>A interval on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:attribute name="begin" type="xs:unsignedLong" use="required"/>
    <xs:attribute name="end" type="xs:unsignedLong" use="required"/>
</xs:complexType>
```

Updated XSD position type:

```
<xs:complexType name="position">
    <xs:attribute name="position" type="xs:long" use="required"/>
</xs:complexType>
```

Updated XSD interval type:

```
<xs:complexType name="interval">
    <xs:annotation>
        <xs:documentation>A interval on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:attribute name="begin" type="xs:long" use="required"/>
    <xs:attribute name="end" type="xs:long" use="required"/>
</xs:complexType>
```

a.4) Resulting sequences for mutation

Current XSD baseLocation type:

```
<xs:complexType name="baseLocation">
    <xs:annotation>
        <xs:documentation>A location on a sequence. Both begin and end can be a defined position, a fuzzy position, or undetermined.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:sequence>
            <xs:element name="startStatus" type="mif:cvType">
                <xs:annotation...>
            </xs:element>
            <xs:choice minOccurs="0">
                <xs:element name="begin" type="mif:position">
                    <xs:annotation...>
                </xs:element>
                <xs:element name="beginInterval" type="mif:interval">
                    <xs:annotation...>
                </xs:element>
            </xs:choice>
        </xs:sequence>
        <xs:sequence>
            <xs:element name="endStatus" type="mif:cvType">
                <xs:annotation...>
            </xs:element>
            <xs:choice minOccurs="0">
                <xs:element name="end" type="mif:position">
                    <xs:annotation...>
                </xs:element>
                <xs:element name="endInterval" type="mif:interval">
                    <xs:annotation...>
                </xs:element>
            </xs:choice>
        </xs:sequence>
        <xs:element name="isLink" type="xs:boolean" default="false" minOccurs="0">
            <xs:annotation...>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

Updated XSD baseLocation type:

```
<xs:complexType name="baseLocation">
    <xs:annotation>
        <xs:documentation>A location on a sequence. Both begin and end can be a defined position, a fuzzy position, or undetermined.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:sequence>
            <xs:element name="startStatus" type="mif:cvType">
                <xs:annotation...>
            </xs:element>
            <xs:choice minOccurs="0">
                <xs:element name="begin" type="mif:position">
                    <xs:annotation...>
                </xs:element>
                <xs:element name="beginInterval" type="mif:interval">
                    <xs:annotation...>
                </xs:element>
            </xs:choice>
        </xs:sequence>
        <xs:sequence...>
        <xs:element name="isLink" type="xs:boolean" default="false" minOccurs="0">
            <xs:annotation...>
        </xs:element>
        <xs:element name="resultingSequence" type="mif:resultingSequenceType1" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The resultingSequence gives some information about the mutated sequence.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD resultingSequence type (proposal 1):

```
<xs:complexType name="resultingSequenceType1">
    <xs:annotation>
        <xs:documentation>Description of the mutated or transformed interactor sequence portion.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="sequence" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>The mutated or transformed sequence portion.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="xref" type="mif:xref" minOccurs="0">
            <xs:annotation>
                <xs:documentation>External cross reference to a genetic variation database such as dbSNP.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD resultingSequence type (proposal 2):

```
<xs:complexType name="resultingSequenceType2">
    <xs:annotation>
        <xs:documentation>Description of the mutated or transformed interactor sequence portion</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="originalSequence" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>The original sequence portion.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="newSequence" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>The mutated or transformed sequence portion.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="xref" type="mif:xref" minOccurs="0">
            <xs:annotation>
                <xs:documentation>External cross reference to a genetic variation database such as dbSNP.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.5) Participant stoichiometry

Current XSD participant type:

```
<xs:complexType name="participant">
    <xs:annotation>
        <xs:documentation>A molecule participating in an interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence...>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

New XSD stoichiometry, maxStoichiometry and minStoichiometry attributes (proposal 1):

```
<xs:complexType name="participant">
    <xs:annotation...>
    <xs:sequence...>
    <xs:attribute name="id" type="xs:int" use="required"/>
    <xs:attribute name="stoichiometry" type="xs:decimal" use="optional">
        <xs:annotation>
            <xs:documentation>The stoichiometry of the participant.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="minStoichiometry" type="xs:decimal" use="optional">
        <xs:annotation>
            <xs:documentation>The minimum stoichiometry observed for this participant.
                This attribute should always be used in pair with the maxStoichiometry attribute.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="maxStoichiometry" type="xs:decimal" use="optional">
        <xs:annotation>
            <xs:documentation>The maximum stoichiometry observed for this participant.
                This attribute should always be used in pair with the minStoichiometry attribute.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
```

New XSD choice element in participant (proposal 2):

```
<xs:complexType name="participant">
    <xs:annotation>
        <xs:documentation>A molecule participating in an interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice>
            <xs:annotation...>
            <xs:element name="interactorRef" type="xs:int"...>
            <xs:element name="interactor" type="mif:interactor"...>
            <xs:element name="interactionRef" type="xs:int"...>
        </xs:choice>
        <xs:element name="participantIdentificationMethodList" type="mif:participantIdentificationMethodList" minOccurs="0"/>
        <xs:element name="biologicalRole" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentalRoleList" type="mif:experimentalRoleList" minOccurs="0"/>
        <xs:element name="experimentalPreparationList" type="mif:experimentalPreparationList" minOccurs="0"/>
        <xs:element name="experimentalInteractorList" type="mif:experimentalInteractorList" minOccurs="0"/>
        <xs:element name="featureList" type="mif:featureList" minOccurs="0"/>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
        <xs:choice minOccurs="0">
            <xs:annotation>
                <xs:documentation>Description of the participant stoichiometry. Can be a single stoichiometry or a stoichiometry range.</xs:documentation>
            </xs:annotation>
            <xs:element name="stoichiometry" type="mif:stoichiometryType">
                <xs:annotation>
                    <xs:documentation>The stoichiometry of the participant.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="stoichiometryRange" type="mif:stoichiometryRangeType">
                <xs:annotation>
                    <xs:documentation>The stoichiometry range of the participant.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

New XSD stoichiometry and stoichiometryRange element (proposal 2):

```
<xs:complexType name="stoichiometryType">
    <xs:annotation>
        <xs:documentation>The mean value for the participant stoichiometry.
        </xs:documentation>
    </xs:annotation>
    <xs:attribute name="value" type="xs:decimal" use="required">
        <xs:annotation>
            <xs:documentation>The participant stoichiometry value</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>

<xs:complexType name="stoichiometryRangeType">
    <xs:annotation>
        <xs:documentation>The stoichiometry range of a participant.
        </xs:documentation>
    </xs:annotation>
    <xs:attribute name="minValue" type="xs:decimal" use="required">
        <xs:annotation>
            <xs:documentation>The minimum stoichiometry value</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="maxValue" type="xs:decimal" use="required">
        <xs:annotation>
            <xs:documentation>The maximum stoichiometry value</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
```

a.6) Feature detection method list

Current XSD feature having one feature detection method:

```
<xs:complexType name="feature">
    <xs:annotation>
        <xs:documentation>A feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Experimental method used to identify the feature. A setting here overrides the global setting given in the experimentDescription.
                    External controlled vocabulary. </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Updated XSD feature having a feature detection method list:

```
<!--Feature description-->
<xs:complexType name="feature">
    <xs:annotation>
        <xs:documentation>A feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureDetectionMethodList" type="mif:featureDetectionMethodList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The list of feature detection methods used to detect and confirm this feature. </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
<xs:complexType name="featureDetectionMethodList">
    <xs:annotation>
        <xs:documentation>A list of feature detection methods used to identify and confirm the feature.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="featureDetectionMethod" type="mif:cvType" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>Experimental method used to identify the feature. A setting here overrides the global setting given in the experimentDescription.
                    External controlled vocabulary. </xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.7) Feature and interaction dependency type (resulting-ptm)

Current XSD feature having one feature type:

```
<xs:complexType name="feature">
    <xs:annotation>
        <xs:documentation>A feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureType" type="mif:cvType" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Description and classification of the feature. This element is controlled by the PSI-MI controlled vocabulary "feature", root term id MI:011
            </xs:annotation>
        </xs:element>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Updated XSD feature having a featureTypeList (proposal 1):

```
<xs:complexType name="feature">
    <xs:annotation>
        <xs:documentation>A feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureTypeList" type="mif:featureTypeList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The list of feature types describing the feature.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
<xs:complexType name="featureTypeList">
    <xs:annotation>
        <xs:documentation>A list of feature types describing the feature.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="featureType" type="mif:cvType" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>Description and classification of the feature. This element is controlled by the PSI-MI controlled vocabulary "feature",
                    root term id MI:0116.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

Updated XSD feature having a single featureType and an interactionDependencyType and an interactionEffectType elements (proposal 2):

```
<xs:element name="featureRangeList">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="featureRange" type="mif:baseLocation" maxOccurs="unbounded">
                <xs:annotation>
                    <xs:documentation>Location of the feature on the sequence of the interactor. One feature may have more than one featur
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="interactionDependencyType" type="mif:cvType" minOccurs="0">
    <xs:annotation>
        <xs:documentation>Describes the impact of the interaction on the feature. (Ex: resulting-ptm, ...)</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="interactionEffectType" type="mif:cvType" minOccurs="0">
    <xs:annotation>
        <xs:documentation>Describes the impact of the feature on the interaction. (Ex: prerequisite-ptm,...)</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="attributeList" type="mif:attributeList" minOccurs="0">
    <xs:annotation>
        <xs:documentation>Semi-structured additional description of the data contained in the entry.</xs:documentation>
    </xs:annotation>
</xs:element>
```

a.8) Interactor sets

Current XSD interactorList :

```
<xs:complexType name="interactorList">
    <xs:annotation>
        <xs:documentation>List of all interactors occurring in the entry</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="interactor" type="mif:interactor" minOccurs="0"
                    maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>A molecule object in its native state, as described in databases.
                </xs:documentation>
                <xs:documentation>Usage: A protein interactor must contain an xref to UniProt and
                    NCBI-GI where possible.
                </xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

Current XSD participant element :

```
<xs:complexType name="participant">
    <xs:annotation>
        <xs:documentation>A molecule participating in an interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice>
            <xs:annotation>
                <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry, fully describes an interactor, or references another i
            </xs:annotation>
            <xs:element name="interactorRef" type="xs:int"...>
            <xs:element name="interactor" type="mif:interactor"...>
            <xs:element name="interactionRef" type="xs:int"...>
        </xs:choice>
        <xs:element name="participantIdentificationMethodList" type="mif:participantIdentificationMethodList" minOccurs="0"/>
        <xs:element name="biologicalRole" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentalRoleList" type="mif:experimentalRoleList" minOccurs="0"/>
        <xs:element name="experimentalPreparationList" type="mif:experimentalPreparationList" minOccurs="0"/>
        <xs:element name="experimentalInteractorList" type="mif:experimentalInteractorList" minOccurs="0"/>
        <xs:element name="featureList" type="mif:featureList" minOccurs="0"/>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Update XSD interactorList with interactorSet (proposal 1):

```
<xs:complexType name="interactorList">
    <xs:annotation>
        <xs:documentation>List of all interactors occurring in the entry</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="interactor" type="mif:interactor">
                <xs:annotation>
                    <xs:documentation>A molecule object in its native state, as described in databases.
                    </xs:documentation>
                    <xs:documentation>Usage: A protein interactor must contain an xref to UniProt and
                        NCBI-GI where possible.
                    </xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="interactorSet" type="mif:interactorSet">
                <xs:annotation>
                    <xs:documentation>A set of molecule objects in their native state, as described in databases and either of them can interact.
                    </xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
</xs:complexType>
```

New XSD interactorSet element in the interactorList (proposal 1):

```
<xs:complexType name="interactorSet">
    <xs:annotation>
        <xs:documentation>Describes a set of molecules that can interact. It is not an interaction because the interactors defined in the interactorSet
            are not interacting with each other.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="interactorSetType" type="mif:cvType">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="interactorCandidateList" type="mif:interactorCandidateList">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
<xs:complexType name="interactorCandidateList">
    <xs:annotation>
        <xs:documentation>A set of interactors that are not interacting with each other but either of them can interact with another molecule.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="interactorRef" type="xs:int" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>References an interactor described in the interactorList of the entry
                </xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD interactorSet in the participant (proposal 1):

```
<xs:complexType name="interactorSet2">
    <xs:annotation>
        <xs:documentation>Describes a set of molecules that can interact. It is not an interaction because the interactors defined in the interactorSet
            are not interacting with each other.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="interactorSetType" type="mif:cvType"...>
        <xs:element name="interactorCandidateList" type="mif:interactorCandidateList2">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="interactorCandidateList2">
    <xs:annotation>
        <xs:documentation>A set of interactors that are not interacting with each other but either of them can interact with another molecule.</xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:annotation>
            <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry or fully describes an interactor. </xs:documentation>
        </xs:annotation>
        <xs:element name="interactorRef" type="xs:int" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>References an interactor described in the interactorList of the entry</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="interactor" type="mif:interactor" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>Fully describes an interactor</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:choice>
</xs:complexType>
```

New XSD interactorSet (proposal 2):

```
<xs:complexType name="interactorSet2">
    <xs:annotation>
        <xs:documentation>Describes a set of molecules that can interact. It is not an interaction because the interactors defined in the interactorSet
            are not interacting with each other.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="interactorSetType" type="mif:cvType">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="interactorCandidateList" type="mif:interactorCandidateList2">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="interactorCandidateList2">
    <xs:annotation>
        <xs:documentation>A set of interactors that are not interacting with each other but either of them can interact with another molecule.</xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:annotation>
            <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry or fully describes an interactor. </xs:documentation>
        </xs:annotation>
        <xs:element name="interactorRef" type="xs:int" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>References an interactor described in the interactorList of the entry</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="interactor" type="mif:interactor" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>Fully describes an interactor</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:choice>
</xs:complexType>
```

Updated XSD participant with interactorSet (proposal 1 and 2):

```
<xs:complexType name="participant">
    <xs:annotation>
        <xs:documentation>A molecule participating in an interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice>
            <xs:annotation>
                <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry, fully describes an interactor, or references another i
            </xs:annotation>
            <xs:element name="interactorRef" type="xs:int"...>
            <xs:element name="interactor" type="mif:interactor"...>
            <xs:element name="interactorSet" type="mif:interactorSet2">
                <xs:annotation>
                    <xs:documentation>Describes a set of interactors</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="interactionRef" type="xs:int"...>
        </xs:choice>
        <xs:element name="participantIdentificationMethodList" type="mif:participantIdentificationMethodList" minOccurs="0"/>
        <xs:element name="biologicalRole" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentalRoleList" type="mif:experimentalRoleList" minOccurs="0"/>
        <xs:element name="experimentalPreparationList" type="mif:experimentalPreparationList" minOccurs="0"/>
        <xs:element name="experimentalInteractorList" type="mif:experimentalInteractorList" minOccurs="0"/>
        <xs:element name="featureList" type="mif:featureList" minOccurs="0"/>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
        <xs:choice minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Updated XSD participantList with participantSet (proposal 3):

```
<xs:complexType name="participantList2">
    <xs:annotation>
        <xs:documentation>A list of molecules participating in this interaction. An interaction has one
            (intramolecular), two (binary), or more (n-ary, complexes) participants.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:choice minOccurs="1" maxOccurs="unbounded">
            <xs:element name="participant" type="mif:participant"/>
            <xs:element name="participantSet" type="mif:participantSet"/>
        </xs:choice>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="participantSet">
    <xs:annotation>
        <xs:documentation>Describes a set of molecules that can interact and each of them has different participant properties such as features, stoichiometry, etc.. It is no
            are not interacting with each other.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="participantSetType" type="mif:cvType">
            <xs:annotation>
                <xs:documentation>The type of the interactorSet, e.g. open set. This element is controlled by the PSI-MI controlled vocabulary.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="participantCandidateList" type="mif:participantCandidateList">
            <xs:annotation>
                <xs:documentation>List of participant candidates.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Allows semi-structured additional annotation of the participantSet.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="participantCandidateList">
    <xs:annotation>
        <xs:documentation>A set of participants/entities that are not interacting with each other but either of them can interact with another molecule and have their own pro
    </xs:annotation>
    <xs:sequence>
        <xs:element name="participant" type="mif:participant" maxOccurs="unbounded" minOccurs="1">
            <xs:annotation>
                <xs:documentation>Participant candidate
                </xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.9) Complexes

a.9.1) Feature ranges referring to a participant

Updated XSD baseLocation type with participantRef:

```
<xs:complexType name="baseLocation">
    <xs:annotation>
        <xs:documentation>A location on a sequence. Both begin and end can be a defined position, a fuzzy position, or undetermined.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:sequence>
            <xs:element name="startStatus" type="mif:cvType"...>
            <xs:choice minOccurs="0">
                <xs:element name="begin" type="mif:position"...>
                <xs:element name="beginInterval" type="mif:interval"...>
            </xs:choice>
        </xs:sequence>
        <xs:sequence>
            <xs:element name="endStatus" type="mif:cvType"...>
            <xs:choice minOccurs="0">
                <xs:element name="end" type="mif:position"...>
                <xs:element name="endInterval" type="mif:interval"...>
            </xs:choice>
        </xs:sequence>
        <xs:element name="isLink" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="participantRef" type="xs:int" minOccurs="0">
            <xs:annotation>
                <xs:documentation>References a participant described in the entry</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.9.2) Complexes as an 'abstract' interaction

Current XSD interactionList element:

```
<xs:complexType name="interactionList">
    <xs:annotation>
        <xs:documentation>List of interactions</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="interaction" type="mif:interaction" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
```

Current XSD interaction element:

```
<xs:complexType name="interaction">
    <xs:annotation>
        <xs:documentation>A molecular interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice minOccurs="0">
            <xs:annotation>
                <xs:documentation>Either refer to an already defined availability statement in this entry or insert description.<
            </xs:annotation>
            <xs:element name="availabilityRef" type="xs:int"...>
            <xs:element name="availability" type="mif:availability"...>
        </xs:choice>
        <xs:element name="experimentList" type="mif:experimentList" minOccurs="0"/>
        <xs:element name="participantList" type="mif:participantList"/>
        <xs:element name="inferredInteractionList" type="mif:inferredInteractionList" minOccurs="0"/>
        <xs:element name="interactionType" type="mif:cvType" minOccurs="0" maxOccurs="unbounded"...>
        <xs:element name="modelled" type="xs:boolean" minOccurs="0"...>
        <xs:element name="intraMolecular" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="negative" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"/>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="imexId" type="xs:string" use="optional"/>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

Current XSD participant element:

```
<xs:complexType name="participant">
    <xs:annotation>
        <xs:documentation>A molecule participating in an interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice>
            <xs:annotation>
                <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry, fully de
            </xs:annotation>
            <xs:element name="interactorRef" type="xs:int"...>
            <xs:element name="interactor" type="mif:interactor"...>
            <xs:element name="interactionRef" type="xs:int"...>
        </xs:choice>
        <xs:element name="participantIdentificationMethodList" type="mif:participantIdentificationMethodList" minOccurs="0"/>
        <xs:element name="biologicalRole" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentalRoleList" type="mif:experimentalRoleList" minOccurs="0"/>
        <xs:element name="experimentalPreparationList" type="mif:experimentalPreparationList" minOccurs="0"/>
        <xs:element name="experimentalInteractorList" type="mif:experimentalInteractorList" minOccurs="0"/>
        <xs:element name="featureList" type="mif:featureList" minOccurs="0"/>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Current XSD feature element:

```
<xs:complexType name="feature">
    <xs:annotation>
        <xs:documentation>A feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

Current XSD inferredInteractionList, inferredInteraction and inferredParticipant elements:

```
<xs:complexType name="inferredInteractionList">
    <xs:annotation>
        <xs:documentation>Describes inferred interactions, usually combining data from more than one experiment.
            Examples: 1: Show the topology of binary interactions within a complex. 2: Interaction inferred from
            multiple experiments which on their own would not support the interaction. Example: A-B in experiment 1,
            B-C- in experiment 2, A-C is the inferred interaction.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="inferredInteraction" type="mif:inferredInteraction" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="inferredInteraction">
    <xs:sequence>
        <xs:element name="participant" type="mif:inferredInteractionParticipant" minOccurs="2" maxOccurs="unbounded"/>
        <xs:element name="experimentRefList" type="mif:experimentRefList" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="inferredInteractionParticipant">
    <xs:annotation>
        <xs:documentation>Participant of the inferred interaction.</xs:documentation>
    </xs:annotation>
    <xs:choice>
        <xs:element name="participantRef" type="xs:int"/>
        <xs:element name="participantFeatureRef" type="xs:int"/>
    </xs:choice>
</xs:complexType>
```

Updated XSD interaction element with optional experimentList , organism and interactorType elements (proposal 1):

```
<xs:complexType name="interaction">
    <xs:annotation>
        <xs:documentation>A molecular interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Name(s) of the interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="xref" type="mif:xref" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Interaction database ID</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:choice minOccurs="0"...>
        <xs:element name="experimentList" type="mif:experimentList" minOccurs="0"/>
        <xs:element name="participantList" type="mif:participantList"/>
        <xs:element name="inferredInteractionList" type="mif:inferredInteractionList" minOccurs="0"/>
        <xs:element name="interactionType" type="mif:cvType" minOccurs="0" maxOccurs="unbounded"...>
        <xs:element name="modelled" type="xs:boolean" minOccurs="0"...>
        <xs:element name="intraMolecular" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="negative" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"/>
        <xs:element name="parameterList" type="mif:parameterList" minOccurs="0"/>
        <xs:element name="organism" type="mif:bioSource" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The organism where this interaction naturally occurs (organism of a stable complex, etc.).</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="interactorType" type="mif:cvType" minOccurs="0">
            <xs:annotation>
                <xs:documentation>The interactor type if the interaction is used as a participant of another interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="imexId" type="xs:string" use="optional"/>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

Updated XSD interactionList element with optional modelledInteraction (proposal 2):

```
<xs:complexType name="interactionList">
    <xs:annotation>
        <xs:documentation>List of interactions</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:choice maxOccurs="unbounded">
            <xs:element name="interaction" type="mif:interaction"/>
            <xs:element name="modelledInteraction" type="mif:modelledInteraction">
                <xs:annotation>
                    <xs:documentation>Abstract interaction describing a stable complex, allosteric interaction , etc...
                    </xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
</xs:complexType>
```

New XSD modelledInteraction element (proposal 2):

```
<xs:complexType name="modelledInteraction">
    <xs:annotation>
        <xs:documentation>An 'abstract' molecular interaction - e.g - stable complexes, allosteric interaction, ....</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="1">
            <xs:annotation>
                <xs:documentation>Name(s) of the interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="participantList" type="mif:modelledParticipantList"/>
        <xs:element name="bindingFeatureList" type="mif:bindingFeatureList" minOccurs="0"/>
        <xs:element name="interactionType" type="mif:cvType" minOccurs="0">
            <xs:annotation>
                <xs:documentation>External controlled vocabulary characterising the interaction type,
                    for example "physical interaction".</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="intraMolecular" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="organism" type="mif:bioSource" minOccurs="1"...>
        <xs:element name="interactorType" type="mif:cvType" minOccurs="1">
            <xs:annotation>
                <xs:documentation>The interactor type for this complex when it is used as a participant of another interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="confidenceList" type="mif:modelledConfidenceList" minOccurs="0"/>
        <xs:element name="parameterList" type="mif:modelledParameterList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

New XSD modelledConfidence element (proposal 2: a confidence with bibRef and not experimentRef):

```
<xs:complexType name="modelledConfidenceList">
    <xs:annotation>
        <xs:documentation>A list of confidence values for a complex or abstract interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="confidence" type="mif:modelledConfidence" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="modelledConfidence">
    <xs:annotation>
        <xs:documentation>A confidence value for a complex or other 'abstract' interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="type" type="mif:openCvType"/>
        <xs:element name="value">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:minLength value="1"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="bibRef" type="mif:bibref" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
```

New XSD modelledParameter element (proposal 2: a parameter with bibRef and not experimentRef):

```
<xs:complexType name="modelledParameterList">
    <xs:annotation>
        <xs:documentation>Lists parameters which are relevant for the complex/abstract interaction, e.g. kinetics.</xs:documentatio
    </xs:annotation>
    <xs:sequence>
        <xs:element name="parameter" type="mif:modelledParameter" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="modelledParameter">
    <xs:complexContent>
        <xs:extension base="mif:parameterBase">
            <xs:sequence>
                <xs:element name="bibRef" type="mif:bibref" minOccurs="0">
                    <xs:annotation>
                        <xs:documentation>Reference to the publication (or publications) where this parameter has been shown.
                        </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="uncertainty" type="xs:decimal" use="optional"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="parameterList"...>
```

New XSD modelledParticipant element (proposal 2: a participant without experimental details):

```
<xs:complexType name="modelledParticipantList">
    <xs:annotation>
        <xs:documentation>A list of molecules participating in a modelled interaction. A complex has one
            (homo-dimers), two (binary), or more (complexes) participants.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="participant" type="mif:modelledParticipant" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="modelledParticipant">
    <xs:annotation>
        <xs:documentation>A molecule participating in a modelled interaction (complex, allsoteric interaction, ...).</xs:documentat
    </xs:annotation>
    <xs:sequence>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:choice>
            <xs:annotation>
                <xs:documentation>Description of the Interactor. Refers to an already defined interactor in this entry, fully descr
            </xs:annotation>
            <xs:element name="interactorRef" type="xs:int"...>
            <xs:element name="interactor" type="mif:interactor"...>
            <xs:element name="modelledInteractionRef" type="xs:int"...>
        </xs:choice>
        <xs:element name="biologicalRole" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureList" type="mif:modelledFeatureList" minOccurs="0"/>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

New XSD modelledFeature element (proposal 2: a feature without experimental details):

```
<xs:complexType name="modelledFeatureList">
    <xs:annotation>
        <xs:documentation>Sequence features relevant for the modelled interaction, for example binding domains.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="feature" type="mif:modelledFeature" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="modelledFeature">
    <xs:annotation>
        <xs:documentation>A biological feature, e.g. domain, on a sequence.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Names for the feature, e.g. SH3 domain.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="featureType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureRangeList"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"/>
</xs:complexType>
```

New XSD bindingFeatureList, bindingFeatures elements (proposal 2: list the binding sites to describe topology of a complex):

```
<xs:complexType name="bindingFeatureList">
    <xs:annotation>
        <xs:documentation>Show the topology of interactions within a complex.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="bindingFeature" type="mif:bindingFeatures" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="bindingFeatures">
    <xs:sequence>
        <xs:element name="participantFeatureRef" type="xs:int" minOccurs="2" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
```

New XSD cooperativeEffectList element :

```
<xs:complexType name="cooperativeEffectList">
    <xs:annotation>
        <xs:documentation>A list of cooperative effects this interaction has on subsequent interactions, either through
    </xs:annotation>
    <xs:sequence>
        <xs:choice minOccurs="1" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>The cooperative mechanism can be either allostery or pre-assembly.</xs:documentation>
            </xs:annotation>
            <xs:element name="allostery" type="mif:allostery">
                <xs:annotation>
                    <xs:documentation>In case the cooperative mechanism is allostery.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="preassembly" type="mif:cooperativeEffectType">
                <xs:annotation>
                    <xs:documentation>In case the cooperative mechanism is pre-assembly.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
</xs:complexType>
```

New XSD cooperativeEffectType :

```
<xs:complexType name="evidenceType">
    <xs:annotation>
        <xs:documentation>List of experimental methods and corresponding publication from which this cooperative effect has been inferred.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="bibref" type="mif:bibref" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Publication describing the experiments from which this cooperative effect has been inferred.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="evidenceMethodList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Experimental methods from which this cooperative effect has been inferred.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="evidenceMethod" type="mif:cvType" minOccurs="1" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD allostery element :

```
<xs:complexType name="allostery">
    <xs:annotation>
        <xs:documentation>In case the cooperative mechanism is allostery.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="mif:cooperativeEffectType">
            <xs:sequence>
                <xs:element name="allostericMolecule" type="xs:int" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Refers to the participant that is allosterically regulated.</xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:choice minOccurs="1">
                    <xs:element name="allostericEffector" type="xs:int" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>Refers to the participant that elicits an allosteric response in an allosteric molecule upon binding to that mol
                        </xs:annotation>
                    </xs:element>
                    <xs:element name="allostericModification" type="xs:int" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>Refers to the modification (feature) that elicits an allosteric response in an allosteric molecule.</xs:document
                        </xs:annotation>
                    </xs:element>
                </xs:choice>
                <xs:element name="allostericMechanism" type="mif:cvType" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Indicates the type of changes that occur in an allosteric molecule upon allosteric modification or binding of an all
                    </xs:annotation>
                </xs:element>
                <xs:element name="allosteryType" type="mif:cvType" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Indicates the chemical relationship between the ligands whose binding is allosterically coupled. This element is com
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

```
<xs:complexType name="cooperativeEffectType">
    <xs:annotation>
        <xs:documentation>A cooperative effect an interaction has on a subsequent interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="cooperativityEvidenceList">
            <xs:annotation...>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="cooperativityEvidenceDescription" type="mif:evidenceType" minOccurs="1" maxOccurs="unbounded">
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="affectedInteractionList">
            <xs:annotation...>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="affectedInteraction" type="xs:int" minOccurs="1" maxOccurs="unbounded">
                        <xs:annotation>
                            <xs:documentation>Refers to the model interaction that is affected by the current model interaction.</xs:documentation>
                        </xs:annotation>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="cooperativeEffectOutcome" type="mif:cvType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Indicates whether the cooperative effect is positive (either induced or enhanced) or negative
                    (either inhibited or abrogated). This element is controlled by the PSI-MI controlled vocabulary "cooperative effect outcome"
                    root term, id MI:1153.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="cooperativeEffectResponse" type="mif:cvType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Indicates how the cooperative mechanism affects another interaction. This element is controlled by the PSI-MI
                    controlled vocabulary "cooperative effect response" root term, id MI:.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="cooperativeEffectValue" type="xs:decimal" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Quantification of the cooperative effect. The ratio of a binding or a catalytic parameter of the affected interaction in
                    the absence versus presence of the affecting interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
</xs:complexType>
```

a.10) Cooperative interactions

Updated XSD modelledInteraction element with cooperativeEffectList.
Same changes if cooperativeEffectList is in interaction element :

```
<xs:complexType name="modelledInteraction">
    <xs:annotation...>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="1"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="participantList" type="mif:modelledParticipantList"/>
        <xs:element name="bindingFeatureList" type="mif:bindingFeatureList" minOccurs="0"/>
        <xs:element name="interactionType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="intraMolecular" type="xs:boolean" default="false" minOccurs="0"...>
        <xs:element name="organism" type="mif:bioSource" minOccurs="1"...>
        <xs:element name="interactorType" type="mif:cvType" minOccurs="0"...>
        <xs:element name="confidenceList" type="mif:modelledConfidenceList" minOccurs="0"/>
        <xs:element name="parameterList" type="mif:modelledParameterList" minOccurs="0"/>
        <xs:element name="cooperativeEffectList" type="mif:cooperativeEffectList" minOccurs="0"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

New XSD cooperativeEffectList element :

```
<xs:complexType name="cooperativeEffectList">
    <xs:annotation>
        <xs:documentation>A list of cooperative effects this interaction has on subsequent interactions, either through
    </xs:annotation>
    <xs:sequence>
        <xs:choice minOccurs="1" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>The cooperative mechanism can be either allostery or pre-assembly.</xs:documentation>
            </xs:annotation>
            <xs:element name="allostery" type="mif:allostery">
                <xs:annotation>
                    <xs:documentation>In case the cooperative mechanism is allostery.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="preassembly" type="mif:cooperativeEffectType">
                <xs:annotation>
                    <xs:documentation>In case the cooperative mechanism is pre-assembly.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
</xs:complexType>
```

New XSD cooperativeEffectType :

```
<xs:complexType name="cooperativeEffectType">
    <xs:annotation>
        <xs:documentation>A cooperative effect an interaction has on a subsequent interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="cooperativityEvidenceList">
            <xs:annotation...>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="cooperativityEvidenceDescription" type="mif:evidenceType" minOccurs="1" maxOccurs="unbounded">
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="affectedInteractionList">
            <xs:annotation...>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="affectedInteraction" type="xs:int" minOccurs="1" maxOccurs="unbounded">
                        <xs:annotation>
                            <xs:documentation>Refers to the model interaction that is affected by the current model interaction.</xs:documentation>
                        </xs:annotation>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="cooperativeEffectOutcome" type="mif:cvType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Indicates whether the cooperative effect is positive (either induced or enhanced) or negative
                    (either inhibited or abrogated). This element is controlled by the PSI-MI controlled vocabulary "cooperative effect outcome"
                    root term, id MI:1153.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="cooperativeEffectResponse" type="mif:cvType" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Indicates how the cooperative mechanism affects another interaction. This element is controlled by the PSI-MI
                    controlled vocabulary "cooperative effect response" root term, id MI:.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="cooperativeEffectValue" type="xs:decimal" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Quantification of the cooperative effect. The ratio of a binding or a catalytic parameter of the affected interaction in
                    the absence versus presence of the affecting interaction.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
</xs:complexType>
```

New XSD allostery element :

```
<xs:complexType name="allostery">
    <xs:annotation>
        <xs:documentation>In case the cooperative mechanism is allostery.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="mif:cooperativeEffectType">
            <xs:sequence>
                <xs:element name="allostericMolecule" type="xs:int" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Refers to the participant that is allosterically regulated.</xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:choice minOccurs="1">
                    <xs:element name="allostericEffector" type="xs:int" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>Refers to the participant that elicits an allosteric response in an allosteric molecule upon binding to that mol
                        </xs:annotation>
                    </xs:element>
                    <xs:element name="allostericModification" type="xs:int" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>Refers to the modification (feature) that elicits an allosteric response in an allosteric molecule.</xs:document
                        </xs:annotation>
                    </xs:element>
                </xs:choice>
                <xs:element name="allostericMechanism" type="mif:cvType" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Indicates the type of changes that occur in an allosteric molecule upon allosteric modification or binding of an all
                    </xs:annotation>
                </xs:element>
                <xs:element name="alloseryType" type="mif:cvType" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>Indicates the chemical relationship between the ligands whose binding is allosterically coupled. This element is con
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

New XSD evidenceType :

```
<xs:complexType name="evidenceType">
    <xs:annotation>
        <xs:documentation>List of experimental methods and corresponding publication from which this cooperative effect has been inferred.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="bibref" type="mif:bibref" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Publication describing the experiments from which this cooperative effect has been inferred.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="evidenceMethodList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Experimental methods from which this cooperative effect has been inferred.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="evidenceMethod" type="mif:cvType" minOccurs="1" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.11) Dynamic interactions and variable experiment conditions

Current XSD experimentDescription element :

```
<xs:complexType name="experimentDescription">
    <xs:annotation>
        <xs:documentation>Describes one set of experimental parameters.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"/>
        <xs:element name="bibref" type="mif:bibref"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="interactionDetectionMethod" type="mif:cvType"...>
        <xs:element name="participantIdentificationMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0"...>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

Updated XSD experimentDescription element :

```
<xs:complexType name="experimentDescription">
    <xs:annotation>
        <xs:documentation>Describes one set of experimental parameters.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="names" type="mif:names" minOccurs="0"/>
        <xs:element name="bibref" type="mif:bibref"...>
        <xs:element name="xref" type="mif:xref" minOccurs="0"...>
        <xs:element name="hostOrganismList" type="mif:hostOrganismList" minOccurs="0"/>
        <xs:element name="interactionDetectionMethod" type="mif:cvType"...>
        <xs:element name="participantIdentificationMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="featureDetectionMethod" type="mif:cvType" minOccurs="0"...>
        <xs:element name="variableParameterList" type="mif:variableParameterList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>A list of variable parameters used in this experiment - eg - variable concentration of a specific
            </xs:annotation>
        </xs:element>
        <xs:element name="confidenceList" type="mif:confidenceList" minOccurs="0"...>
        <xs:element name="attributeList" type="mif:attributeList" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Semi-structured additional description of the experiment.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required"...>
</xs:complexType>
```

New XSD variableParameterList and variableParameter elements:

```
<xs:complexType name="variableParameterList">
    <xs:annotation>
        <xs:documentation>A list of variable parameters used in this experiment - eg - variable concentration of a specific drug.</x
    </xs:annotation>
    <xs:sequence>
        <xs:element name="variableParameter" type="mif:variableParameter" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="variableParameter">
    <xs:annotation>
        <xs:documentation>Describes one variable parameter and its values in this experiment  - eg - variable concentration of a spe
    </xs:annotation>
    <xs:sequence>
        <xs:element name="description" type="xs:string">
        <xs:annotation>
            <xs:documentation>Free description of the variable parameter (such as cell cycle, PMA treatment, ...).</xs:documentation
        </xs:annotation>
        </xs:element>
        <xs:element name="unit" type="mif:cvType" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Unit of the variable parameter values.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="variableValueList" type="mif:variableValueList">
            <xs:annotation>
                <xs:documentation>List of the different values for this specific variableParameter in this experiment.</xs:documenta
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD variableValueList and variableValue elements:

```
<xs:complexType name="variableValueList">
    <xs:annotation>
        <xs:documentation>List of the different values for a specific variableParameter in a specific experiment.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="variableValue" type="mif:variableValue" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="variableValue">
    <xs:annotation>
        <xs:documentation>A value for a specific variableParameter in a specific experiment - eg - the concentration of a specific c
    </xs:annotation>
    <xs:sequence>
        <xs:element name="value" type="xs:string">
            <xs:annotation>
                <xs:documentation>Free description of the variable value. It can be numerical value or qualitative value depending c
            </xs:annotation>
        </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int" use="required">
        <xs:annotation>
            <xs:documentation>Unique numerical identifier for this variableValue so an interaction can refer to it later. The id has
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="order" type="xs:int" use="optional">
        <xs:annotation>
            <xs:documentation>Optional numerical order attribute to give an explicit order for a variableValue in the variableValueL
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
```

New XSD experimentalVariableValueList, variableValues and variableValueRef elements in the interaction element:

```
<xs:complexType name="experimentalVariableValueList">
    <xs:annotation>
        <xs:documentation>A list of experimental parameter/condition values for which the interaction occurs.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="variableValues" minOccurs="1" maxOccurs="unbounded" type="mif:variableValues">
            <xs:annotation>
                <xs:documentation>A set of experimental parameter/conditions values applied together and for which this interaction occurs.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="variableValues">
    <xs:annotation>
        <xs:documentation>A set of experimental parameter/conditions values applied together and for which this interaction occurs.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="variableValueRef" minOccurs="1" maxOccurs="unbounded" type="xs:int">
            <xs:annotation>
                <xs:documentation>The reference to the id of the variableValue described in the variableParameterList/variableParameter/variableParameterValueL
                    of the experiment.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

a.12) Causal relationships

New XSD causalRelationshipList in the interaction element (proposals 1 and 2):

```
<xs:element name="causalRelationshipList" type="mif:causalRelationshipList" minOccurs="0">
    <xs:annotation>
        <xs:documentation>A list of causal relationships involving participants in this interaction</xs:documentation>
    </xs:annotation>
</xs:element>
```

```
<xs:complexType name="causalRelationshipList">
    <xs:annotation>
        <xs:documentation>A list of causal relationships involving participants in a specific interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="causalRelationship" type="mif:causalRelationship" minOccurs="0" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>The causal relationship: participant source -> causality statement -> participant target.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD causalRelationshipList in the participant element (proposals 3 and 4):

```
<xs:element name="causalRelationshipList" type="mif:causalRelationshipList2" minOccurs="0">
    <xs:annotation>
        <xs:documentation>A list of causal relationships involving this participant in this interaction</xs:documentation>
    </xs:annotation>
</xs:element>
```

```
<xs:complexType name="causalRelationshipList2">
    <xs:annotation>
        <xs:documentation>A list of causal relationships involving a specific participant in a specific interaction.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="causalRelationship" type="mif:causalRelationship2" minOccurs="0" maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation>The causal relationship: causality statement -> participant target.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD causalRelationship (proposal 1):

```
<xs:complexType name="causalRelationship">
    <xs:annotation>
        <xs:documentation>The causal relationship: participant source -> causality statement -> participant target..</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="participantSourceRef" type="xs:int" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Refers to the participant that is the source of the causality statement.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="causalityStatement" type="mif:cvType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>The causality statement. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxxx.</xs:docu
            </xs:annotation>
        </xs:element>
        <xs:element name="participantTargetRef" type="xs:int" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Refers to the participant that is the target of the causality statement.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD causalRelationship (proposal 2):

```
<xs:complexType name="causalRelationshipBis">
    <xs:annotation>
        <xs:documentation>The causal relationship: participant source -> causality statement -> participant target..</xs:documentation>
    </xs:annotation>
    <xs:attribute name="participantSourceRef" type="xs:int" use="required">
        <xs:annotation>
            <xs:documentation>Refers to the participant that is the source of the causality statement.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="causalityStatement" use="required">
        <xs:annotation>
            <xs:documentation>The causality statement. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxxx.</xs:document
        </xs:annotation>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="causalityStatementAc" use="required">
        <xs:annotation>
            <xs:documentation>The causality statement MI identifier. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxx
        </xs:annotation>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="participantTargetRef" type="xs:int" use="required">
        <xs:annotation>
            <xs:documentation>Refers to the participant that is the target of the causality statement.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
```

New XSD causalRelationship (proposal 3):

```
<xs:complexType name="causalRelationship2">
    <xs:annotation>
        <xs:documentation>The causal relationship: causality statement -> participant target..</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="causalityStatement" type="mif:cvType" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>The causality statement. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxxx.</xs:docu
            </xs:annotation>
        </xs:element>
        <xs:element name="participantTargetRef" type="xs:int" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>Refers to the participant that is the target of the causality statement.</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
```

New XSD causalRelationship (proposal 4):

```
<xs:complexType name="causalRelationshipBis2">
    <xs:annotation>
        <xs:documentation>The causal relationship: causality statement -> participant target..</xs:documentation>
    </xs:annotation>
    <xs:attribute name="causalityStatement" use="required">
        <xs:annotation>
            <xs:documentation>The causality statement. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxxx.</xs:document
        </xs:annotation>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="causalityStatementAc" use="required">
        <xs:annotation>
            <xs:documentation>The causality statement MI identifier. This element is controlled by the PSI-MI controlled vocabulary "causality statement", root term id MI:xxx
        </xs:annotation>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="participantTargetRef" type="xs:int" use="required">
        <xs:annotation>
            <xs:documentation>Refers to the participant that is the target of the causality statement.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
```