ResNet Exchange Format

Specification version 1.3 (Target PS version: 9.0)

ELSEVIER

ResNet Exchange Format (RNEF) is designed to provide a common portable external representation for MedScan output, PathwayStudio Pathway Save/Load operations, ResNet Database dumps, and output of third-party tools. The format is XML-based, so RNEF files can be validated and processed by a multitude of XML tools (e.g. XSLT engine)[1]. Some appendices to this document describe backward-compatible extensions to the format specified in the main section below.

The format is structured as a sequence of self-contained units called *resnets* (when assembled together, they form a single ResNet database). Each resnet is a collection of *attributes*, *nodes* (entities), and *controls/links*, connecting these nodes (relations between entities). Resnets represent closed fragments of a large Biological Association Network (BAN); each connection in a proper resnet connects nodes described inside this resnet (no dangling links). Individual resnets are merged on load using external identities of their nodes (URNs).

Besides what is given by the closure property, individual resnets have no inherent semantics; their role in the system can vary from representing arbitrary subnets produced by MedScan from a single input sentence to groups and pathways. A particular role can be assigned by tagging the resnet with optional **name=**, **type=**, and **urn=** attributes as described below.

Each resnet in a RNEF file is represented by a **<resnet>** element. All resnets are written sequentially in the arbitrary order inside the top-level **<batch>** element:

```
<batch>
    <resnet …> … </resnet>
    <resnet …> … </resnet>
    …
</batch>
```

RNEF's **<batch>** element has no XML attributes and no semantics besides grouping; several RNEF files can be assembled into one by putting all resnets into a single batch.

RNEF's XML format can be extended beyond the formal definition presented here. To be on a safe side, all tools importing RNEF files should ignore XML attributes they do not understand. The import tools should also be ready to ignore unknown elements following the standard ones within the **<resnet>** element. Likely extensions to RNEF are pathway visualization and graphic layout data.

A **<batch>** element may be annotated with an optional **<properties>** element which, if present, should precede all **<resnet>** elements. A **<properties>** element has no XML attributes; it contains a list of **<attr>** elements specifying the properties. Each **<attr>** element has two required XML attributes **name=** and **value=**

```
<batch>
    <properties>
        <attr name="Notes" value="My first pathway collection" />
        …
    </properties>
    <resnet …> … </resnet>
    …
</batch>
```

The list of supported property names for batch-level properties depends on a particular application. For backward compatibility with RNEF specifications prior to 1.2, programs producing RNEF output should not use batch-level properties. Programs reading RNEF should be ready to read these properties, ignoring those which they can't interpret.

A **<resnet>** element can have XML attributes defining its role. Common situations and roles are listed in the Table 1 below. Note that if no XML attributes are present, it is up to the import tool to decide the appropriate role for a resnet. Tool designers can decide what to do with roles that do not meet their expectations; a general guideline is to always signal the user if some information in the RNEF file has been ignored or interpreted in nonstandard way.

<resnet> attributes	role
<resnet>	(assigned by the importer)
<resnet name="caspase" type="FunctionalClass" urn="…">	Functional class named caspase
<resnet name="NF-kappaB" type="Complex" urn="…">	Complex named NF-kappaB identified by a URN
<resnet name="MAPK" type="Pathway">	Pathway named MAPK
<resnet name="Receptors" type="Group">	Group named Receptors
<resnet name="Kinases" type="Group" urn="…">	Group named Kinases identified by a URN

Table 1. Resnet XML attributes and corresponding resnet roles.

The optional **urn=** attribute is used to provide a globally unique identifier for a collection of nodes. The same URN is used for a collection of nodes as **<resnet>** and for a node, identifying the collection as a single entity. Globally identified groups are usually used to represent protein annotations of ontology categories. See Appendix E for details.

Each **\<resnet\>** element contains an optional **\<properties\>** element, followed by required **\<nodes\>** and **\<controls\>** elements:

```
<batch>
    <resnet>
        <nodes> … </nodes>
        <controls> … </controls>
    </resnet>
    …
    <resnet name="MAPK">
        <properties> … </properties>
        <nodes> … </nodes>
        <controls> … </controls>
    </resnet>
    …
</batch>
```

A resnet-level **\<properties\>** element specifies name/value pairs describing Pathways and Groups; it is omitted in MedScan-generated RNEF files where **\<resnet\>**s correspond to input sentences. A resnet-level **\<properties\>** element has no XML attributes; it contains a list of **\<attr\>** elements specifying the properties. Each **\<attr\>** element has two required XML attributes **name=** and **value=**, and an optional XML attribute **index=**:

```
<batch>
    <resnet name="MAPK" type="Pathway">
        <properties>
            <attr name="Notes" value="My first pathway" />
            <attr name="Author" value="A.U.Thor' index="1" />
            <attr name="Author" value="C.O.Author" index="2" />
            …
        </properties>
        …
    </resnet>
    …
</batch>
```

An **\<attr\>** element has no sub-elements; its **name=** XML attribute is selected from an extendable vocabulary, as shown in Table 2. If needed, more than one **\<attr\>** element with the same **name=** attribute can be specified; such **\<attr\>s** may have optional **index=** XML attributes assigning them integer "indices" to distinguish them (indices have to be non-negative; no duplicate indices are allowed in sibling **\<attr\>** elements with the same name). On the resnet properties level, \<**attr**\> element indices have no implied semantics; programs accepting RNEF input can either ignore them or use them for presentation purposes, for example to impose a sorting order.

The required **<nodes>** element follows the optional **<properties>** element (if any). Each **<nodes>** element contains a list of **<node>** elements describing BAN entities (proteins, complexes, small molecules, etc.).

<attr name="…"	value="…" />
Source	Optional; an organization/database source (used for Pathways)
Organism	Optional; a source organism (Pathways)
ParentGroups	Optional; for nested Groups, a list of parent groups each prefixed with /, e.g. '/group1/group2'
Notes	Optional; a free-form commentary (Groups/Pathways)
...	(Other; See Appendix B)
X-name	Optional; extended attribute added by a third party

Table 2. Standard **<resnet>** properties.

```
<batch>
    <resnet name="MAPK" type="Pathway">
        <properties> … </properties>
        <nodes>
            <node local_id="N1" urn="urn:agi-llid:9191">
                <attr name="NodeType" value="Protein" />
                <attr name="Name" value="POLR2D" />
                …
            </node>
            <node local_id="N2" urn="urn:agi-llid:2353">
                …
            </node>
            …
        </nodes>
        …
    </resnet>
    …
</batch>
```

Each **<node>** element has two required XML attributes **local_id=** and **urn=**. Local IDs are unique labels given to each node/control within a **<resnet>** (control labels are described below). By an informal convention, node labels are formed by appending 'N' to the index of a node within the **<resnet>**. The **urn=** attribute specifies the globally unique Universal Resource Name to the node. URNs used by Ariadne Genomics are described in the "Ariadne URNs" document.

The contents of a **<node>** element are node properties (**<attr>** elements), structured similarly to the **<resnet>** properties described above; the only difference is in the

vocabulary of property names / property values (see Table 3). The node property vocabulary is also extendable, and more property names may be added by third parties.

<attr name="…"	value="…" />

NodeType	Required; one of {
	   Protein,
	   CellObject,
	   Complex,
	   Enzyme,
	   SmallMol,
	   CellProcess,
	   Treatment,
	   FunctionalClass,
	   Disease,
	   Pathway [2],
	   Group [3],
	   Ontology [4],
	   Folder [5],
	 }

Name	Required; a display name for the node (short)
...	(Other; See Appendices A, B)
X-*name*	Optional; extended property added by a third party

Table 3. Standard **<node>** properties.

The required **<controls>** element follows the **<nodes>** element. Each **<controls>** element contains a list of **<control>** elements describing BAN Controls (relations between BAN Nodes):

```
<batch>
    <resnet name="Foobar" type="Pathway">
        <properties> … </properties>
        <nodes> … </nodes>
        <controls>
            <control local_id="L1">
                <link type="in" ref="N1" />
                <link type="out" ref="N2" />

                …
                <attr name="NodeType" value="Protein" />
                <attr name="Name" value="POLR2D" />

                …
            </control>
```

---

[2] This node type is reserved for "non-atomic" objects; its use is limited to certain specific contexts
[3] This node type is reserved for "non-atomic" objects; its use is limited to certain specific contexts
[4] This node type is reserved for "non-atomic" objects; its use is limited to certain specific contexts
[5] This node type is reserved; its use is limited to certain specific contexts (see Appendix G)

```
             …
          </controls>
      </resnet>
</batch>
```

Each **<control>** element has a required XML attribute **local_id=**. Local IDs are unique labels given to each node/control within a **<resnet>**. By informal convention, control labels are formed by appending 'L' to the index of a control within the **<resnet>** ('M' is used for non-binary controls). This convention guarantees that each node and each control has a unique local ID.

The contents of a **<control>** element are one or more *primary* links to other nodes/controls (**<link>** elements), followed by zero or more *regulatory* links to other nodes (**<xlink>** elements), and followed by zero or more properties (**<attr>** elements).

Primary links (**<link>** elements) have two required XML attributes **type=** and **ref=**. The **type=** attribute is one of {**in-out**, **in**, **out**}. Link type specifies the role of a particular "target" (another local node or control, specified by the **ref=** attribute) in the relation described by the control. Most controls describe binary relations with two links corresponding to two participants of the relation; directionless relation like binding is described by two **in-out** links, while directional ones use one **in** and one **out** link. Controls, describing relations with more than one participant will have the corresponding number of links; link types should correspond to the roles of the targets in the relation. The **ref=** attribute of a **<link>** element should be equal to the **local_id=** attribute of exactly one node or control in the given resnet; self-references and indirect cycles are not allowed.

Regulatory links (**<xlink>** elements) usually describe "external" factors affecting the relation described by the control. These factors (e.g. substances working as catalyzers) are not treated as main/required participants of the relation. Most control types allow any number of regulatory links.

Each **<xlink>** element has four required XML attributes: **type=**, **ref=**, **effect=**, and **link_id=**. The **type=** attribute is one of {**in-out**, **in**, **out**} and describes the direction of regulation (**in** means that the linked factor affects the main relation, other types are more-or-less open to interpretation). The **effect=** attribute is one of {**negative**, **unknown**, **positive**}; it describes whether activation or inhibition took place (e.g. **type="in" effect="positive"** means "activated-by"). The **link_id=** attribute is an alphanumerical ID uniquely identifying this xlink among other xlinks in the same **<control>** element (IDs are used to refer to this particular xlink if such a need arises in the future). The **ref=** attribute of an **<xlink>** element should be equal to the **local_id=** attribute of exactly one node in the given resnet.

Unlike **<link>** elements, **<xlink>** element can have their own properties. The properties are described by **<attr>** elements inside **<xlink>** element; these properties are ignored by Pathway Studio, their main use being to provide lossless conversion between control-of-control and regulatory-link representation models.

Control properties (**<attr>** elements) are structured similarly to the **<resnet>** and **<node>** properties described above, but use control-specific vocabulary of property names / property values (see Table 4). The control property vocabulary is also extendable, and more property names can be added by third parties.

In context of a control, **index=** XML attributes are used for a specific purpose – to distinguish sets of properties which came from a single source (piece of evidence). All properties specific to the first source are assigned index value 1, second – 2, etc.

Some of the control types and property names described below have two names – a historical one, retained for backward compatibility with RNEF 1.2 (marked *obsolete*) and a new one, introduced in RNEF 1.3 (marked *new*); in RNEF 1.3 many mechanism-like properties are remapped to a single Mechanism property. To be on a safe side, any tool accepting RNEF on input should treat both old and new names as synonyms. Note that RNEF 1.2 is still a default output format for many MedScan pipeline tools.

<attr name="…"	value="…" />

ControlType	Required; one of {
	  UnknownRegulation *(obsolete)*, **Regulation** *(new)*,
	  ExpressionControl *(obsolete)*, **Expression** *(new)*,
	  Binding,
	  PromoterBinding,
	  MolTransport,
	  MolSynthesis,
	  CellObjectControl, *(obsolete)*
	  ProtModification,
	  DirectRegulation,
	  ChemicalReaction,
	  UnknownRelation,
	  Correlation,
	  MemberOf [6]
	}

Effect	Optional; one of {
	  positive,
	  negative,
	  unknown
	}

ExpressionMechanism *(obsolete)*	Optional; one of {
	  transcriptional,
	  posttranscriptional
	}

TransportType *(obsolete)*	Optional; one of { **unknown**,

---

[6] See Appendix E

	import, **export** }

Mechanism	Optional;
	 one of {
	 promoter binding,
	 direct interaction,
	 phosphorylation,
	 desumoylation,
	 deglycosylation,
	 sumoylation,
	 arginylation,
	 acetylation,
	 deacetylation,
	 demethylation,
	 dephosphorylation,
	 methylation,
	 prenylation,
	 farnesylation,
	 geranylgeranylation,
	 GPI anchor linkage,
	 myristylation,
	 palmitoylation,
	 ribosylation,
	 alkylation,
	 biotinylation,
	 dealkylation,
	 flavinylation,
	 nitrosylation,
	 hydroxylation,
	 sulfatation,
	 glycosylation,
	 expression,
	 …
	 }

ModificationType *(obsolete)*	Optional; one of { **phosphorylation**, **desumoylation**, **deglycosylation**, **sumoylation**, **arginylation**, **acetylation**, **deacetylation**, **demethylation**, **dephosphorylation**, **methylation**,

	prenylation, … }
COCType *(obsolete)*	Optional; one of { **biogenesis**, **assembly**, **disassembly**, **movement** }
mref	Optional; PubMed reference in a form *PubMedID***:***SentenceNum*
msrc	Optional; source sentence (within abstract/article identified by **mref**, if **mref** is given)
Relationship	(Optional; See *Appendix E*)
…	(Other; See *Appendix B)*
X-*name*	Optional; extended property added by a third party

Table 4. Standard **<control>** properties.

ControlType	Property	LinkTypes (prim./sec.)
PromoterBinding	Effect[1]	in,out / *any secondary*
Binding	-	in-out* / *any secondary*
Expression *(new)*, ExpressionControl *(obsolete)*	ExpressionMechanism *(obsolete)*, Mechanism *(new)*, Effect[1]	in,out / *any secondary*
MolTransport	TransportType *(obsolete)*, Mechanism *(new)*, Effect[1]	in,out / *any secondary*
MolSynthesis	Effect[1]	in,out / *any secondary*
ProtModification	ModificationType *(obsolete)*, Mechanism *(new)*, Effect[1]	in,out / *any secondary*
DirectRegulation	Effect[1], Mechanism	in,out / *any secondary*
UnknownRelation		in-out* /
Regulation *(new)*, UnknownRegulation *(obsolete)*	Effect, Mechanism	in,out / *any secondary*
ChemicalReaction	Mechanism	in*,out*,in-out* /
CellObjectControl *(obsolete)*	COCType *(obsolete)*, Effect[1]	in,out / *any secondary*
Correlation		in-out*
MemberOf	Relationship	in*,out*

Table 5. Control-specific properties.

Appendix A: Optional Node Properties

RNEF Nodes may have multiple properties carrying miscellaneous information and links to popular databases. All of these properties except **Description** and **Notes** may be present multiple times (multiple Aliases etc.).

Optional RNEF Node properties are listed in the table below.

<attr name="…"	value="…" />

---

[1] Effect property is optional; if makes sense and omitted, it is taken as "unknown". ProtModification and PromoterBinding can be assigned positive or negative effect as a result of Control merge procedure, when they are merged with congruent UnknownRegulation. For compatibility with older versions of PS, it is recommended to generate explicit Effect="unknown" wherever optional Effect is allowed.

Alias	Alternative common name. Format is the same as for the **Name** property (short string with no implied relation to any nomenclature). Case-sensitive. Examples: "p53" (while Name can be "TP53")
Source	Used to identify the original source of node identity. Nodes unified from multiple sources may have many Source properties. Values currently in use: "ResNet" "Curated" "Jubilant" "ERGO from Integrated Genomics" "KEGG" "BIND" "DIP"
Organism	Latin name of the organism. Case-insensitive (word capitalization is preferred). Values currently in use: "Homo Sapiens" "Mus Musculus" "Rattus Norvegicus" "Arabidopsis Thaliana" "Caenorhabditis Elegans" "Drosophila Melanogaster" "Saccharomyces Cerevisiae"
CAS ID	CAS Registry Number for a chemical as defined by CAS Registry (www.cas.org). A CAS Registry Number is a numeric identifier that can contain up to 9 digits, divided by hyphens into 3 parts; the right digit is a check digit used to verify the validity and uniqueness of the entire number. Examples: "58-08-2" (CAS Registry Number for caffeine)
HGNC ID	Numerical ID in the HUGO Gene Nomenclature Committee's database (www.gene.ucl.ac.uk). Examples: "11998" (tumor protein p53)

Hugo ID	*Obsolete*; alias for HGNC ID described above. RNEF tools compatible with this specification should produce only HGNC ID properties, but accept both HGNC ID and Hugo ID properties.
HUGO Symbol	Gene symbol as assigned by HUGO Gene Nomenclature Committee (www.gene.ucl.ac.uk). The "symbol" is a unique series of Latin (upper case in human) letters and Arabic numbers which usually is no longer than six characters in length. Approved symbols should be used if they exist. Case-sensitive. Examples: "BRCA1" (symbol for "breast cancer, early onset 1" gene)
PubMed ID	This property refers to articles that describe the biological or conceptual entity identified by the node. It is used, for example, to annotate proteins with the PMIDs of the important articles describing their function. This property SHOULD NOT be used to refer to arbitrary mentioning of the entity in the literature; the references should be only to the articles having the entity as primary subject. Values of PubMed ID properties are numerical IDs with no leading zeroes ([1-9][0-9]*)
Swiss-Prot Accession	Protein identifier as assigned by the UniProt/Swiss-Prot Protein Knowledgebase (www.ebi.ac.uk). Accession numbers consist of 6 alphanumerical characters in the following format: [O,P,Q] [0-9] [A-Z, 0-9] [A-Z, 0-9] [A-Z, 0-9] [0-9] An accession number (AC) is assigned to each sequence upon inclusion into UniProt. Accession numbers are stable from release to release. Each entry has one primary AC and may have optional secondary ACs. Examples: "P08251" (Sodium/potassium-transporting ATPase beta-1 chain protein)
Swiss-Prot ID	Protein Accession number or entry name in UniProt/Swiss-Prot Protein Knowledgebase

	(www.ebi.ac.uk). RNEF tools compatible with this specification should *only* generate this property for entry names (of form X_Y where X is a menemonic code of at most 4 alphanumeric characters and Y is a mnemonic species identification code of at most 5 alphanumeric characters representing the biological source of the protein), but accept *both* accession numbers and entry names. Preferred property for Swiss-Prot accession numbers is **Swiss-Prot Accession**.
GenBank ID	Accession Number (ACCN) or GenInfo Identifier (GI) for a protein as assigned by NCBI's GenBank (www.ncbi.nih.gov/Genbank). Accession Number is the unique identifier for a sequence record. An accession number applies to the complete record and is usually a combination of a letter(s) and numbers, such as a single letter followed by five digits (e.g., "U12345") or two letters followed by six digits (e.g., "AF123456"). Records from the RefSeq database of reference sequences have a different accession number format that begins with two letters followed by an underscore bar and six or more digits, for example "NM_002111". Accession numbers do not change, even if information in the record is changed at the author's request. To identify a particular version, it can be appended to ACCN, e.g. "U12345.6". Case-sensitive. The GI system of sequence identifiers runs parallel to the accession.version system. If the protein sequence changes in any way, it will receive a new GI number, and the version will be incremented by one. GI is purely numerical identifier. Examples: "NM_002111", "AAA98665.2", "1293613"
RGD ID	Rat Genome database ID (http://rgd.mcw.edu) . Must be a positive integer. Example: "3889" (tumor protein p53)
Entrez GeneID	LocusID or GeneID in NCBI's LocusLink/Entrez Gene database (www.ncbi.nih.gov/entrez). LocusLink is phased out, but its LocusIDs are used as-is in the new Entrez

	Gene database. The values are positive integers. Example: "7157" (tumor protein p53)
Unigene ID	Cluster ID in NCBI's UniGene database (www.ncbi.nih.gov/UniGene). IDs are formed from short denotation for organism and numerical cluster ID, separated by dot. Case-sensitive. Example: "Hs.408312" (p53 cluster)
MGI ID	Mouse Genome Informatics ID (http://www.informatics.jax.org/). Must be a positive integer. Example: "98834" (tumor protein p53)
PIR ID	Accession number in PIR PSD database (http://pir.georgetown.edu/). Alphanumerical, case-sensitive. Examples: "A24849", "S38568"
Microarray ID	Used for the microarray probe identifier for Affymetrix or any other chip. Alphanumerical with underscores; case-sensitive Example: "1457623_x_at" (tumor protein p53)
GO ID	Gene Ontology (www.geneontology.org) 7-digit numerical group identifier without the "GO:" prefix. Use it to identify the GO group for the node. Example: "0004791" (thioredoxin-disulfide reductase)
EC Number	Identificator of an anzyme assigned by IUBMB Enzyme Commission (www.chem.qmw.ac.uk/iubmb/enzyme/), without the "EC" prefix. The identifer itself is four numbers separated by periods. Example: "3.4.11.4" (tripeptide aminopeptidase)
OMIM ID	The numerical identifier in Online Inheritance in Man (OMIM) database without the "OMIM" prefix (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM) Example: "114500" (tumor protein p53)

KEGG ID	KEGG identifier as used in KGML format (www.genome.jp/kegg). KEGG IDs are formed from short denotation for source database or organism and local ID in the database, separated by colon. For some organisms ("hsa", "mmu", "rno") database-dependent part is LocusLink ID; "ec" database uses EC Numbers. Case-sensitive. Examples: "hsa:7157" (Tumor protein p53) "cpd:C00074" "ec:1.14.21.3"
STKE Component ID	Component ID in Science's STKE Pathway database (stke.sciencemag.org). Component IDs uniquely identify entities in STKE Component Library. IDs have the form "stkecm_CMC_N", where N is a positive number. Case-sensitive. Example: "stkecm_CMC_7615" (Tumor protein p53)
PubChem CID	This is used to annotate a small molecule with its corresponding PubChem compound record. The value should be a numerical IDs with no leading zeroes ([1-9][0-9]*)
PubChem SID	This is used to annotate small molecules with their corresponding PubChem substance records. Values are is numerical IDs with no leading zeroes ([1-9][0-9]*)
URL	Web link to a page dedicated to the entity described by the node. Syntax follows usual URL conventions. Case-sensitive. Example (folded): "http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene

	&cmd=Retrieve&dopt=Graphics&list_uids=7157"
Description	One-line plain text description of the entity. Example: "Tumor protein p53 (Li-Fraumeni syndrome)"
Notes	Multi-line plain text description of the entity. Example: "Tumor protein p53, a nuclear protein, plays an essential role in the regulation of cell cycle, specifically in the transition from G0 to G1. It is found in very low levels in normal cells, however, in a variety of transformed cell lines, it is expressed in high amounts, and believed to contribute to transformation and malignancy. p53 is a DNA-binding protein containing DNA-binding, oligomerization and transcription activation domains."
Cell Localization	Is used by layout by cell localization algorithm in PathwayAssist as default value for Localization in Pathway for any node. Has to have one of the following values: "Golgi" "Mitochondria" "Endoplasmic reticulum" "Nucleus" "Cytoplasm" "Plasma membrane"
GO Cellular Component	Name of Gene Ontology Cellular component group; case-sensitive. Example: "axonemal heterotrimeric kinesin-II complex"
GO Molecular Function	Name of Gene Ontology Molecular Function group; case-sensitive. Example: "epidermal growth factor binding"

GO Biological Process	Name of Gene Ontology Biological Process group; case-sensitive. Example: "drinking behavior"
MeSH Heading	MeSH heading for the node taken in its original text form as-is, with original case, spaces, commas etc. The best way to ensure correct case and punctuation is to copy the heading verbatim from the "MeSH Heading" field of the NLM entry page for the term, e.g. http://www.nlm.nih.gov/cgi/mesh/2005/MB_cgi?mode=&term=Breast+Neoplasms,+Male&field=entry. Example: "Leukemia, Monocytic, Acute"

Table 6. Optional **<node>** properties.

Appendix A1: Optional Control Properties

RNEF Controls may have multiple properties carrying additional information on a specific reference (a single source of evidence for a control). These properties should be given as **<attr>** elements with the value of **index=** attribute corresponding to the sequential number of the reference (indices can be omitted if there is only one reference for the given control).

Optional RNEF Control properties are listed in the table below.

<attr name="…"	value="…" />
TextRef	Case-sensitive URI (Universal Resource Identifier) for the source fragment. The identification of the fragment within the source is added as #fragment_type:fragment_no suffix. This property is a preferred way to identify the source of the given reference. **Note: *this property will be required in the next version of the RNEF specification.*** PubMed URI Examples: "info:pmid/14617836#abs:9" (9th sentence of the abstract) "info:pmid/14617836#title:1" (1st sentence of the title) "info:pmid/14617836#body:21" (21st sentence of the body text) "info:pmid/14617836#cont:21" (21st sentence of unstructured content) DOI URI Example: "info:doi/10.1023/A:1007582911958#cont:12" File URI Examples: "file:///C:/Users/Joe/Documents/Journal%20Downloads/Web_Batch2.zip/Round4/PMID_10714616.xml:cont:12" "file://laptop314/My%20Documents/Downloads/PMID_10714616.xml#cont:21" HTML URI Example: "http://www.springerlink.com/content/p76945262t55738h/#cont:21"
PMID	PubMed Identifier of the source article. Example: "10714616"
PII	Publisher-specific Item Identifier of the source article. Example: "S0034-89102009005000078"

DOI	Digital Object Identifier of the source article. Example: "10.1023/A:1007582911958"
PMC	PubMed Central Identifier of the source article. Example: "PMC2796602"
Title	Title of the source article
Authors	Authors of the source article. Individual authors are separated by semicolons; authors names are given as last name, comma, intitial, dot, initial, dot etc. form. Example: "Jones,P.; Smith,J.D.;Adams,B.S. Jr"
ISSN	Print ISSN of the source article's journal. Example: "1234-1983"
ESSN	Electronic ISSN of the source article's journal. Example: "7890-1234"
MedlineTA	Medline's Title Abbreviation for the journal name. Example: "J Appl Genet"
PubYear	Year of publication for the source article. Example: "2009"
PubMonth	3-letter abbreviation for the month of publication for the source article. Other formats are possible. Examples: "Jan", "Jan-Feb"
PubDay	Day of publication. Examples: "21", "1"
PubTypes	Semicolon-separated list of PubMed's publication types. Example: "Journal Article;Research Support, Non-U.S. Gov't"
Volume	Source article's journal volume number. Example: "44"
Issue	Source article's journal issue number. Example: "4"
Pages	Source article's pagination as a page number or a range. Other formats are possible. Examples: "92", "443-450"

Organism	Latin name of the organism for which the reference is given. Case-insensitive (word capitalization is preferred). Examples: "Homo Sapiens", "Mus Musculus"
Organ	The organ for which the reference is given. Case-insensitive (lower case is preferred). Examples: "liver", "skin", "lung"
Tissue	The tissue for which the reference is given. Case-insensitive (lower case is preferred). Examples: "alveola", "pulmonary", "mammary"
CellLineName	The name of the cell line for which the reference is given. Case-sensitive. Examples: "RB", "EF", "EBC-1"
CellType	The type of the cell for which the reference is given. Case-insensitive (lower case is preferred). Examples: "fibroblast", "macrophage", "pneumocyte"
TextPath	Digtext's specification for the sequence of steps taken to extract the text fragment used as reference. Examples: "dir(G%3a\Dstributives%20%28MR%29\tmp):file(G%3a\Distributives%20%28MR%29\tmp\p53.eml):MIME-multipart[14]:HTML" "query(c-fos):MIME-multipart[26]:message(http%3a//www.crystalgraphics.com/excel/cfo.main.asp):HTML"
LocatorString	Locscan's locator string for the batch job that produced this reference. Example: "pmquery:c-fos"

Table 7. Optional **<control>** per-reference properties.

Appendix B: Standardized Vocabulary of other Optional Properties

RNEF supports additional properties for resnets, nodes and controls (denoted by ellipsis rows in Tables 2-4). To provide a minimal level of standardization, we provide the list of other properties used by PathwayAssist 2.5. Some of the properties listed below are synonyms, mapped on import (**mref** = **MedLine Reference**, **msrc** = **MedLine Sentence**, **Name** = *XXX* **Name**, etc.).

All properties, not explicitly described in Tables 2-6 are optional. No properties starting with **X-** will be assigned by AGI and included into this specification, so third parties can use them to extend the format.

Localization
State
Pathway Description
Linked Pathway
MedLine Reference
MedLine Sentence
Secondary
DevStage
Proteome Biochemical Function
Proteome Cellular Role
Proteome Organismal Role
Proteome Subcellular Localization
State Description
SwissProt Name
Class
DB Link
Gene
PDB ID
Reference
Chemical Mechanism
KEGG Pathway Map
Group Description
Activity Mechanism
Correlation

Appendix C: RNEF Sample

This is an example of the correct RNEF XML encoding.

```xml
<?xml version='1.0'?>
<!DOCTYPE batch SYSTEM 'resnet.dtd'>
<batch>
 <resnet mref="11965497:3">
  <nodes>
   <node local_id="N1" urn="urn:agi-llid:162989">
    <attr name="NodeType" value="Protein" />
    <attr name="Name" value="162989" />
   </node>
   <node local_id="N2" urn="urn:agi-llid:9191">
    <attr name="NodeType" value="Protein" />
    <attr name="Name" value="POLR2D" />
   </node>
  </nodes>
  <controls>
   <control local_id="L1">
    <link type="in-out" ref="N1" />
    <link type="in-out" ref="N2" />
    <attr name="mref" value="11965497:3" />
    <attr name="TextRef" value="info:pmid/11965497#abs:4" />
    <attr name="msrc" value="ID{162989=FLAME-3} interacts with
ID{9191=DEDD} and ID{8837=c-FLIP} (ID{8837=FLAME-1}) but not with the
other ID{12003119=DED}-containing proteins ID{8772=FADD},
ID{841=caspase-8} or ID{843=caspase-10}." />
    <attr name="ControlType" value="Binding" />
   </control>
  </controls>
 </resnet>
</batch>
```

Appendix D: RNEF DTD

Below is a simplified version of RNEF DTD. A complete machine-readable DTD with comments is also available (resnet.dtd).

```
<!ELEMENT batch (properties?,resnet*)>
<!ELEMENT resnet (properties?,nodes,controls,attachments?)>
<!ELEMENT properties (attr*)>
<!ELEMENT nodes (node*)>
<!ELEMENT controls (control*)>
<!ELEMENT node (attr*)>
<!ELEMENT control (link*,xlink*,attr*)>
<!ELEMENT link EMPTY>
<!ELEMENT xlink (attr*)>
<!ELEMENT attachments ((layout|thumbnail)*)>
<!ELEMENT layout (styles,scene)>
<!ELEMENT styles (style*)>
<!ELEMENT style (attr*)>
<!ELEMENT scene (vobjs,vlinks)>
<!ELEMENT vobjs (vobj*)>
<!ELEMENT vobj (attr*)>
<!ELEMENT vlinks (vlink*)>
<!ELEMENT vlink (attr*)>
<!ELEMENT thumbnail (img)>
<!ELEMENT img EMPTY>
<!ATTLIST resnet name CDATA #IMPLIED>
<!ATTLIST resnet type (Subnet|Pathway|Group) #IMPLIED>
<!ATTLIST resnet urn CDATA #IMPLIED>
<!ATTLIST resnet mref CDATA #IMPLIED>
<!ATTLIST resnet msrc CDATA #IMPLIED>
<!ATTLIST resnet owner CDATA #IMPLIED>
<!ATTLIST resnet refonly CDATA #IMPLIED>
<!ATTLIST node local_id CDATA #REQUIRED>
<!ATTLIST node urn CDATA #REQUIRED>
<!ATTLIST node owner CDATA #IMPLIED>
<!ATTLIST node delete CDATA #IMPLIED>
<!ATTLIST control local_id CDATA #REQUIRED>
<!ATTLIST control owner CDATA #IMPLIED>
<!ATTLIST control delete CDATA #IMPLIED>
<!ATTLIST link type (in|out|in-out) #REQUIRED>
<!ATTLIST link ref CDATA #REQUIRED>
<!ATTLIST xlink type (in|out|in-out) #REQUIRED>
<!ATTLIST xlink ref CDATA #REQUIRED>
<!ATTLIST xlink effect (negative|unknown|positive) #REQUIRED>
<!ATTLIST xlink link_id CDATA #REQUIRED>
<!ATTLIST attr name CDATA #REQUIRED>
<!ATTLIST attr value CDATA #REQUIRED>
<!ATTLIST layout owner CDATA #IMPLIED>
<!ATTLIST styles default_style_sheet CDATA #IMPLIED>
<!ATTLIST style local_id CDATA #REQUIRED>
<!ATTLIST vobj local_id CDATA #REQUIRED>
```

```
<!ATTLIST vobj type
(Node|Control|Link|Clone|Lock|Image|RingImage|Diagram|Text)
#REQUIRED>
<!ATTLIST vobj ref CDATA #IMPLIED>
<!ATTLIST vobj style_ref CDATA #IMPLIED>
<!ATTLIST vlink src_ref CDATA #IMPLIED>
<!ATTLIST vlink dst_ref CDATA #IMPLIED>
<!ATTLIST thumbnail owner CDATA #IMPLIED>
<!ATTLIST img width CDATA #FIXED "256">
<!ATTLIST img height CDATA #FIXED "256">
<!ATTLIST img src CDATA #REQUIRED>
```

Appendix E: Using Resnets to represent membership information

RNEF format supports two methods of representing membership information – *implicit* (standard method for pathways and user groups) and *explicit* (useful for more complex cases, e.g. GO ontology).

There are five common situations involving membership of some form: (1) Pathways as document-like objects consisting of nodes and controls, (2) User groups as document-like objects which are collections of nodes, (3) Ontology groups describing ontology categories potentially having sub-categories and annotated with nodes, (4) protein complex nodes, consisting of individual proteins, and (5) functional class nodes referring to list of proteins which are members of the class.

In spite of the fact that these five kinds of membership (and there may be others) have semantic differences, they all may be represented in RNEF in a regular way. This lowers the effort needed to write processing tools – the tools may share syntactic and structural processing while concentrating on semantic differences as needed by the application.

The *implicit* method represents membership by enclosing all members into a single <**resnet**> element (nodes go into the <**nodes**> section, controls into the <**controls**> section). The <**resnet**> element is given an identity by providing **name=**, **type=**, and optionally **urn=** XML attributes. This method is standard for pathways and user groups (document-like objects which don't have to have URNs and don't require the **urn=** XML attribute). The main part of the specification contains more details and some examples.

A modification of the implicit method is used to represent cases (4) and (5) – membership in functional classes and parts of protein complexes. These and similar cases deal with set-like information annotating nodes with global identity ("atomic entities"), which may participate in pathways, be listed in user groups, etc. The nodes themselves have a regular representation in the form of a <**node**> element, with **urn=** XML attribute providing the identity of the node and <**attr**> elements inside the <**node**> element specifying the properties of the node.

In such cases, membership-like information "natural" for the type of the node can be represented by a single <**resnet**> element which is given the same identity as the corresponding node by providing the **urn=**, **name=**, **type=**, and XML attributes. The identity is based on the URN; type must be the same as the **NodeType** property of the corresponding node, while name is usually the same as the **Name** property of the corresponding node (names are not used for identity purposes).

The modified implicit method is simple and compact, but since it does not specify the "kind" of membership represented by the resnet section, it allows for just one kind per node type. As a result, it is useful only for node types which have a single "natural" membership (e.g. members for functional classes and parts for protein complexes).

The *explicit* method of representing membership information uses *pseudo-controls* to represent the membership relations. Pseudo-controls look like controls but have nothing to do with cause-effect relationships between nodes. A single control type **MemberOf** is used for all pseudo-controls representing membership-like information, with additional semantic information kept in the optional **Relationship** property:

```
<control local_id="C33">
    <attr name="ControlType" value="MemberOf" />
    <attr name="Relationship" value="is-a" />
    <link type="in" ref="G13" />
    <link type="out" ref="G12" />
</control>
```

A single control of type **MemberOf** may have multiple "**in**" links and multiple "**out**" links. It is interpreted as follows: each node referred to by an "**in**" link is a member of all sets referred to by the "**out**" links, in a sense described by the optional **Relationship** property.

Note: GO supports two kinds of relationships between groups, "**is-a**" and "**part-of**". These two are common enough to be used in other situations where relation is not obvious from the type of the participants or needs to be represented explicitly[7].

The "sets" referred to by the "**out**" links are nodes for globally identified entities which have the associated sets with the described relationship. Common examples are nodes of type **Complex** (protein complexes) with the associated set of parts ("**part-of**" relationship), nodes of type **FunctionalClass** (protein functional classes) with the associated set of members ("**is-a**" relationship), and nodes of type **Group** (ontology concepts) with the associated set of sub-concepts and protein members.

As usual, all the nodes referenced by these pseudo-controls need to be present in the same resnet (a **<resnet>** element). This resnet should be "naked", i.e. have no XML attributes or properties. For clarity, the **MemberOf** should not be mixed with controls of other types and should not be used inside resnet sections with explicit types, names, or URNs.

---

[7] Care should be taken not to make unnecessary distinctions; for example, "cat **is a** mammal" vs. "cat is a **member of** the class of mammals" vs. "the set of all cats is a **part of** the set of all mammals" . Concepts should be named by singular words (e.g. "kinase", not "kinases") and **is-a** form should be used by default.

Appendix F: Representing attachments (layouts and thumbnails)

In addition to network information (nodes and controls), Pathway Studio pathways may have associated graphical information that preserves the visual layout of the pathway. RNEF format provides a mechanism of storing the associated information in a form of *attachments*.

The attachments are represented as XML elements enclosed in the optional <**attachments**> element which follows the <**nodes**> and <**controls**> elements:

```
<batch>
    <resnet name="Foobar" type="Pathway">
        <properties> … </properties>
        <nodes> … </nodes>
        <controls> … </controls>
        <attachments>
            <layout> … </layout>
            <thumbnail> … </thumbnail>
        </attachments>
    </resnet>
    …
</batch>
```

The <**layout**> element is described in a separate document.

The <**thumbnail**> element contains a single <**img**> element, patterned after the corresponding HTML element. The <**img**> element has three required XML attributes: **width=**, **height=**, and **src=**. The width and height attributes specify the dimensions of the thumbnail in pixels; currently only 256x256 thumbnails are supported by PS. The **src=** attribute contains a URL of the thumbnail picture. Currently, PS support just one form of this URL, a data: URL (as per RFC 2397, see http://tools.ietf.org/html/rfc2397) with image in PNG form, encoded in Base64. The **src=** attribute can contain newlines, as required to format Base64 stream.

```
<thumbnail>
    <img width="256" height="256"
        src="data:image/png;base64,… "/>
</thumbnail>
```

Appendix G: Using Resnets to represent database dumps

In Pathway Studio Enterprise, RNEF format is used for dumping and loading the contents of the entire database. To support this functionality, RNEF format is extended to describe ownership information and folder structure. These extensions are specific to database dump/restore scenarios and should not be used with regular RNEF processing tools.

In multiuser databases, all objects have an owner (a user). The ownersip information is represented by adding an **owner=** XML attribute to the <**resnet**>, <**node**>, <**layout**>, and <**thumbnail**> elements. Within a single <**attachments**> element no two <**layout**> elements or <**thumbnail**> elements can have the same owner. The value of the **owner=** attribute is a user name (login name).

To represent nested folder structure, the RNEF format is extended with node types for pathways, ontologies, groups, and folders (the corresponding types are listed in Table 3 in italic). These node types do not correspond to network entities (none of them can participate in a pathway, for example), they are used to *map* non-network information into a network form for the database dump/restore scenario. The identity of these nodes are preserved by giving them unique URNs assigned in type-specific URN namespaces.

Folders are represented as nodes of type **Folder**. The explicit method of membership representation is used to specify the contents of folders (subfolders and document-like objects such as pathways). Symbolic links (shortcuts) are represented by **MemberOf** controls with **Relationship** property set to "**symlink**".

Example:

```
<batch>
    <resnet type="Pathway" " name="Foo" urn="urn:agi-pathway:uuid-e8201c72-f6be-
4d09-b084-e9b456c650c7">
        …
    </resnet>
    <resnet>
        <nodes>
            <node local_id="F0" urn="urn:agi-folder:8" owner="Admin">
                <attr name="NodeType" value="Folder"/>
                <attr name="Name" value="Bar"/>
            </node>
            <node local_id="18" urn="urn:agi-pathway:uuid-e8201c72-f6be-4d09-b084-
e9b456c650c7">
                <attr name="NodeType" value="Pathway"/>
                <attr name="Name" value="Foo"/>
            </node>
        </nodes>
        <controls>
            <control local_id="CFE1">
                <attr name="ControlType" value="MemberOf"/>
```

```xml
                        <link type="in" ref="18"/>
                        <link type="out" ref="F0"/>
                </control>
            </controls>
        </resnet>
</batch>
```

Appendix H: Using Resnets to represent deletion information

Normally, RNEF format specifies the entities and relations between them as a collection of data that can be loaded into a database to extend the data already present in the database. When loading a RNEF file into the database, each node (entity) from the RNEF file is checked against the set of nodes in the database, and, depending on the result, either the old node is reused, or a new one is created. For each control (relation) in the RNEF file, either a new control in the database is created, or new references (indexed attributes describing the evidence) are added to the existing control.

In certain situations the goal is not to add to the database, but to find a way to remove the information already present in the database. This appendix describes a compatible extension of the basic RNEF format that supports the deletion scenario as a part of the standard loading procedure.

When loading the data, if a **<resnet>** element has **refonly="true"** XML attribute, no new nodes or controls are created in the database for **<node>** and **<control>** elements inside the **<resnet>** element. Within the scope of such an element the loader operates in "reference-only" mode, locating the existing nodes and controls.

When in "reference-only" mode, the loader looks for **delete="true"** XML attribute on **<node>** and **<control>** elements. These are nodes and controls marked for deletion; if a corresponding node or control is found in the database, it is deleted from the database.

When producing a RNEF file with nodes and controls marked for deletion, one has to make sure that enough information is provided for each node and control to locate them in the database unambiguously. For nodes, the only two pieces of information taken into account are the node's **NodeType** property and its URN (given via **urn=** XML attribute of the <node> element). For controls, the following information defines the control's identity:

1) The **ControlType** property of the control
2) The identities of the control's nodes and types of links connecting them to the control
3) The **Effect** property of the control*
4) The **Mechanism** property of the control*

*Some control types can be configured not to include the values of the Effect and Mechanism properties into the control's identity. See the database configuration documentation for more information on the identity of controls.

Example of the RNEF file with nodes and controls marked for deletion:

```
<batch>

<resnet refonly="true">
   <nodes>
     <node urn="urn:agi-gogroup:0006374" delete="true">
      <attr name="NodeType" value="Group"/>
      <attr name="Name" value="nuclear mRNA splicing via U2-type
spliceosome"/>
     </node>
   </nodes>
  </resnet>

  <resnet refonly="true">
   <nodes>
     <node local_id="N1" urn="urn:agi-llid:338386">
      <attr name="NodeType" value="Protein"/>
      <attr name="Name" value="HSR"/>
     </node>
     <node local_id="N2" urn="urn:agi-gocellproc:0009315">
      <attr name="NodeType" value="CellProcess"/>
      <attr name="Name" value="drug resistance"/>
     </node>
   </nodes>
   <controls>
     <control local_id="L1" delete="true">
      <link type="in" ref="N1"/>
      <link type="out" ref="N2"/>
      <attr name="ControlType" value="Regulation"/>
     </control>:
   </controls>
  </resnet>

</batch>
```

If the database contains all nodes and controls referenced in this example, the loader
should delete one Group node and one Regulation control from the database.