MetaMap 2012 XML Output Explained

The two tables below present MetaMap2012's XML tags listed alphabetically and hierarchically; the two tables contain the same information, only arranged differently.

XML tags are characterized by structure (simple or complex) and number (unique or repeating):

- A *simple* (S) tag is atomic, and consists of only a character string or a number, e.g.,
  <Length>, **<LexCat>**, **<SemType>**, **<Source>**, and **<StartPos>**.
- A *complex* (C) tag contains one or more sub-components, e.g.,
  <Candidate>, **<Mapping>**, **<Negation>**, **<Phrase>**, and **<Utterance>**.
- A *unique* (U) tag occurs only once in the immediately higher-level structure, e.g.,
  <InputMatch>, **<MappingScore>**, **<NegType>**, **<PhraseText>**, and **<PMID>**.
- A *repeating* (R) tag may occur multiple times in the immediately higher-level structure, e.g.,
  <AA>, **<MatchMap>**, **<Option>**, **<SyntaxUnit>**, and **<Token>**.

Certain repeating tags also exist in plural form, denoting a series of one or more of the singular form of the tag, e.g.,
<AAs>, **<AACUIs>**, **<Candidates>**, **<ConceptPIs>**, **<MappingCandidates>**, **<Mappings>**, **<MatchedWords>**, **<MatchMaps>**, **<MMOs>**, **<Negations>**, **<NegConcepts>**,
<NegConcPIs>, **<NegTriggerPIs>**, **<Options>**, **<Phrases>**, **<SemTypes>**, **<Sources>**, **<SyntaxUnits>**, **<Tokens>**, and **<Utterances>**.

Alphabetical listing of current XML tags

Tag	Type	Description
<AAs Count="N"> **<AA>**	CR	All the data generated for an author-defined Acronym/Abbreviation (AA), consisting of • **<AAText>**: the text of the AA, • **<AAExp>**: its expansion, • **<AATokenNum>**: the number of tokens in the AA • **<AALen>**: the character length of the AA • **<AAExpTokenNum>**: the number of tokens in expansion • **<AAExpLen>**: the character length of its expansion, and • **<AACUI>**: any CUIs associated with the expansion of the AA

		The following AA examples will use the text *polymerase chain reaction (PCR)*.
<AACUIs Count="N"> <AACUI>	SR	Any CUIs associated with the expansion of the AA.
<AAExp>	SU	The expansion of the AA (*polymerase chain reaction*)
<AAExpLen>	SU	The character length of the expansion of the AA (25, because *polymerase chain reaction* contains 25 characters)
<AAExpTokenNum>	SU	The number of tokens in the AA expansion (5, because *polymerase chain reaction* contains 5 tokens, including two blank tokens)
<AALen>	SU	The character length of the AA (3, because *PCR* contains 3 characters)
<AAText>	SU	The AA itself (*PCR*)
<AATokenNum>	SU	The number of tokens in the AA (1, because *PCR* contains 1 token)
<Candidates Total="T" Excluded="E" Pruned="P" Remaining="R"> <Candidate>	CR	All the data generated for a candidate concept, including • **<CandidateScore>**: the candidate's negative score, • **<CandidateCUI>**: its CUI, • **<CandidateMatched>**: the candidate matched, • **<CandidatePreferred>**: its preferred name, • **<MatchedWords>**: the text word(s) it matches, • **<MatchMaps>**: the matchmap(s), • **<SemTypes>**: the semantic type(s), • **<IsHead>**: IsHead (yes/no), • **<IsOverMatch>**: IsOverMatch (yes/no), • **<Sources>**: the UMLS source(s), • **<ConceptPIs>**: the positional information, and • **<Status>**: 0/1/2 depending on if candidate is retained/excluded/pruned
<CandidateCUI>	SU	The CUI of the candidate concept
<CandidateMatched>	SU	The candidate concept matched
<CandidatePreferred>	SU	The preferred name of the candidate concept
<CandidateScore>	SU	The negative score of the candidate concept; the computation of this value is explained on pp. 5-9 of MetaMap Evaluation.
<CmdLine>	CU	All the data about the command used to start MetaMap, consisting of • **<Command>**: the actual operating-system call used to start MetaMap, and • **<Option>**: any options passed to MetaMap

<Command>	SU	The actual operating-system call used to start MetaMap
<ConceptPIs Count="N"> <ConceptPI>	CR	The positional information of the concept, consisting of • **<StartPos>**: the 0-based character offset of the concept, counting from the beginning of the input text, and • **<Length>**: the character length of the string
<ConcMatchEnd>	SU	The position within the concept words of the last matching word
<ConcMatchStart>	SU	The position within the concept words of the first matching word
<InputMatch>	SU	The input word(s) making up the syntax unit
<IsHead>	SU	Yes/no value denoting if the candidate concept includes the head of the phrase containing it
<IsOverMatch>	SU	Yes/no value denoting if the candidate concept is an overmatch, i.e., if it contains words on one or both ends that do not match the input text.
<Length>	SU	The character length of the string
<LexCat>	SU	The lexical category of the syntax unit
<LexMatch>	SU	The lexical item(s) matched by the syntax unit
<LexVariation>	SU	The degree of lexical variation between the words in the candidate concept and the words in the phrase; the computation of this value is explained on pp. 2-3 of MetaMap Evaluation.
<MappingCandidates Total="N"> <Candidate>	CU	The candidate concepts participating in a mapping
<Mappings Count="N"> <Mapping>	CR	A set of candidate concepts making up the mapping for the phrase, consisting of • **<MappingScore>**: the negative score of the mapping, and • **<MappingCandidates>**: the candidate concept(s) participating in the mapping.
<MappingScore>	SU	The negative score of the mapping; the computation of this value is explained on pp. 9-10 of MetaMap Evaluation.
<MatchedWords Count="N"> <MatchedWord>	SR	The word(s) in the input text matched by the candidate
<MatchMaps Count="N"> <MatchMap>	CR	A data structure representing • the correspondence of words in the candidate

		concept (**<TextMatchStart>** and **<TextMatchEnd>**) and words in the phrase (**<ConcMatchStart>** and **<ConcMatchEnd>**), and • the lexical variation (**<LexVariation>**) between the words in the candidate concept and the words in the phrase. For example, given the input text *obstructive sleep apnea* and the candidate concept *sleep apnea*, the matching words *sleep* and *apnea* are • the 2nd and 3rd words of the text, and • the 1st and 2nd words of the concept. There is no lexical variation, so the matchmap would therefore be **[[[2,3],[1,2],0]]**. For the candidate concept *sleep apneas*, the MatchMap would be the same, other than having lexical variation of 1 instead of 0.
<MMOs> <MMO>	CR	All the XML output generated for an entire input record or citation, consisting of • **<CmdLine>**: the command used to start MetaMap, • **<AA>**: any acronyms/abbreviation(s) found in the text, • **<Negation>**: any negation(s) found in the text, and • **<Utterances>**: the utterance(s) found in the text
<Negations Count="N"> <Negation>	CR	All the data generated for a negation, including • **<NegType>**: the negation type, • **<NegTrigger>**: the negation trigger, • **<NegTriggerPI>**: the negation trigger's positional information, • **<NegConcepts>**: the negated concept(s), and • **<NegConcPIs>**: the negated concept's StartPos/Length positional information For more information about MetaMap's implementation of NegEx, see the MetaMap09 Release Notes.
<NegConcCUI>	SU	The CUI associated with the negated concept
<NegConcepts Count="N"> <NegConcept>	CR	The negated concept(s), consisting of • **<NegConcCUI>**: the negated concept's CUI, and • **<NegConcMatched>**: the negated concept's name
<NegConcMatched>	SU	The name of the negated concept

<NegConcPIs Count="N"> <NegConcPI>	CR	The StartPos/Length positional information of the negated concept
<NegTrigger>	SU	The negation trigger
<NegTriggerPIs Count="N"> <NegTriggerPI>	CR	The StartPos/Length positional information of the negation trigger
<NegType>	SU	The negation type
<Options Count="N"> <Option>	CR	The option(s) passed to MetaMap, consisting of • **<OptName>**: the option's name, and • **<OptValue>**: the option's value.
<OptName>	SU	The name of the command-line option
<OptValue>	SU	The value of the command-line option (can be null)
<Phrases Count="N"> <Phrase>	CR	The syntactic subcomponent of the utterance, consisting of • **<PhraseText>**: the text of the phrase, • **<SyntaxUnits>**: the syntax unit(s), • **<PhraseStartPos>**: the 0-based character offset of the phrase, counting from the beginning of the input text • **<PhraseLength>**: the character length of the phrase, • **<Candidate>**: any candidate concepts identified in the phrase, and • **<Mapping>**: any mappings created
<PhraseLength>	SU	The character length of the phrase
<PhraseStartPos>	SU	The 0-based character offset of the phrase, counting from the beginning of the input text
<PhraseText>	SU	The text of the phrase
<PMID>	SU	The PubMed ID of the citation containing the utterance
<SemTypes Count="N"> <SemType>	SR	The semantic type(s) of the candidate
<Sources Count="N"> <Source>	SR	The UMLS vocabulary/ies in which the concept was found
<StartPos>	SU	The 0-based character offset of the string, counting from the beginning of the input text
<Status>	SU	0, 1, or 2, representing if candidate was retained (0), excluded (1), or pruned (2)
<SyntaxType>	SU	The syntactic type of the syntax unit (e.g., head, mod, verb, etc.)

<SyntaxUnits Count="N"> <SyntaxUnit>	CR	The syntactic subcomponent of the phrase, consisting of • **<SyntaxType>**: the syntactic type of the syntax unit (e.g., head, mod, verb, etc., • **<LexMatch>**: the lexical item(s), • **<InputMatch>**: the input word(s), • **<LexCat>**: the lexical category, and • **<Tokens>**: the token(s) making up the lexical items
<TextMatchEnd>	SU	The position within the phrase words of the last matching word
<TextMatchStart>	SU	The position within the phrase words of the first matching word
<Tokens Count="N"> <Token>	SR	The tokens making up the lexical items
<Utterances Count="N"> <Utterance>	CR	All the data generated for an utterance, including • **<PMID>**: the utterance's PubMed ID, • **<UttSection>**: the section type (e.g., title or abstract), • **<UttNum>**: the 1-based utterance number within the section, • **<UttText>**: the text of the utterance, • **<UttStartPos>**: the 0-based character offset of the utterance, counting from the beginning of the input text • **<UttLength>**: the length, and • **<Phrases>**: the phrase(s) making up the utterance
<UttLength>	SU	The character length of the utterance
<UttNum>	SU	The 1-based numerical position of the utterance within the section
<UttSection>	SU	The section type (e.g., title or abstract) of the utterance
<UttStartPos>	SU	The 0-based character offset of the utterance, counting from the beginning of the input text
<UttText>	SU	The text of the utterance

Hierarchical listing of current XML tags

Tag	Type	Description

<MMOs> **<MMO>**	CR	All the XML output generated for an entire input record or citation, consisting of • **<CmdLine>**: the command used to start MetaMap, • **<AA>**: any acronyms/abbreviation(s) found in the text, • **<Negation>**: any negation(s) found in the text, and • **<Utterances>**: the utterance(s) found in the text
<CmdLine>	CU	All the data about the command used to start MetaMap, consisting of • **<Command>**: the actual operating-system call used to start MetaMap, and • **<Option>**: any options passed to MetaMap
<Command>	SU	The actual operating-system call used to start MetaMap
<Options Count="N"> **<Option>**	CR	The option(s) passed to MetaMap, consisting of • **<OptName>**: the option's name, and • **<OptValue>**: the option's value.
<OptName>	SU	The name of the command-line option
<OptValue>	SU	The value of the command-line option (can be null)
<AAs Count="N"> **<AA>**	CR	All the data generated for an author-defined Acronym/Abbreviation (AA), consisting of • **<AAText>**: the text of the AA, • **<AAExp>**: its expansion, • **<AATokenNum>**: the number of tokens in the AA • **<AALen>**: the character length of the AA • **<AAExpTokenNum>**: the number of tokens in expansion • **<AAExpLen>**: the character length of its expansion, and • **<AACUI>**: any CUIs associated with

		the expansion of the AA The following AA examples will use the text *polymerase chain reaction (PCR)*.
<AAText>	SU	The AA itself (*PCR*)
<AAExp>	SU	The expansion of the AA (*polymerase chain reaction*)
<AATokenNum>	SU	The number of tokens in the AA (1, because *PCR* contains 1 token)
<AALen>	SU	The character length of the AA (3, because *PCR* contains 3 characters)
<AAExpTokenNum>	SU	The number of tokens in the AA expansion (5, because *polymerase chain reaction* contains 5 tokens, including two blank tokens)
<AAExpLen>	SU	The character length of the expansion of the AA (25, because *polymerase chain reaction* contains 25 characters)
<AACUIs Count="N"><AACUI>	SR	Any CUIs associated with the expansion of the AA.
<Negations Count="N"><Negation>	CR	All the data generated for a negation, including • **<NegType>**: the negation type, • **<NegTrigger>**: the negation trigger, • **<NegTriggerPIs>**: the negation trigger's positional information, • **<NegConcepts>**: the negated concept(s), and • **<NegConcPIs>**: the negated concept's StartPos/Length positional information For more information about MetaMap's implementation of NegEx, see the MetaMap09 Release Notes.
<NegType>	SU	The negation type
<NegTrigger>	SU	The negation trigger
<NegTriggerPIs Count="N"><NegTriggerPI>	CR	The StartPos/Length positional information of the negation trigger
<NegConcepts Count="N"><NegConcept>	CR	The negated concept(s), consisting of • **<NegConcCUI>**: the negated concept's CUI, and • **<NegConcMatched>**: the negated

		concept's name
<NegConcCUI>	SU	The CUI associated with the negated concept
<NegConcMatched>	SU	The name of the negated concept
<NegConcPIs Count="N"> <NegConcPI>	CR	The StartPos/Length positional information of the negated concept
<Utterances Count="N"> <Utterance>	CR	All the data generated for an utterance, including • **<PMID>**: the utterance's PubMed ID, • **<UttSection>**: the section type (e.g., title or abstract), • **<UttNum>**: the 1-based utterance number within the section, • **<UttText>**: the text of the utterance, • **<UttStartPos>**: the 0-based character offset of the utterance, counting from the beginning of the input text • **<UttLength>**: the length, and • **<Phrases>**: the phrase(s) making up the utterance
<PMID>	SU	The PubMed ID of the citation containing the utterance
<UttSection>	SU	The section type (e.g., title or abstract) of the utterance
<UttNum>	SU	The 1-based numerical position of the utterance within the section
<UttText>	SU	The text of the utterance
<UttStartPos>	SU	The 0-based character offset of the utterance, counting from the beginning of the input text
<UttLength>	SU	The character length of the utterance
<Phrases Count="N"> <Phrase>	CR	The syntactic subcomponent of the utterance, consisting of • **<PhraseText>**: the text of the phrase, • **<SyntaxUnits>**: the syntax unit(s), • **<PhraseStartPos>**: the 0-based character offset of the phrase, counting from the beginning of the input text • **<PhraseLength>**: the character length of the phrase,

		• **<Candidate>**: any candidate concepts identified in the phrase, and • **<Mapping>**: any mappings created
<PhraseText>	SU	The text of the phrase
<SyntaxUnits Count="N"> <SyntaxUnit>	CR	The syntactic subcomponent of the phrase, consisting of • **<SyntaxType>**: the syntactic type of the syntax unit (e.g., head, mod, verb, etc., • **<LexMatch>**: the lexical item(s), • **<InputMatch>**: the input word(s), • **<LexCat>**: the lexical category, and • **<Tokens>**: the token(s) making up the lexical items
<SyntaxType>	SU	The syntactic type of the syntax unit (e.g., head, mod, verb, etc.)
<LexMatch>	SU	The lexical item(s) matched by the syntax unit
<InputMatch>	SU	The input word(s) making up the syntax unit
<LexCat>	SU	The lexical category of the syntax unit
<Tokens Count="N"> <Token>	SR	The tokens making up the lexical items
<PhraseStartPos>	SU	The 0-based character offset of the phrase, counting from the beginning of the input text
<PhraseLength>	SU	The character length of the phrase
<Candidates Total="T" Excluded="E" Pruned="P" Remaining="R"> <Candidate>	CR	Total="T" All the data generated for a candidate concept, including • **<CandidateScore>**: the candidate's negative score, • **<CandidateCUI>**: its CUI, • **<CandidateMatched>**: the candidate matched, • **<CandidatePreferred>**: its preferred name, • **<MatchedWords>**: the text word(s) it matches, • **<MatchMaps>**: the matchmap(s), • **<SemTypes>**: the semantic type(s), • **<IsHead>**: IsHead (yes/no), • **<IsOverMatch>**: IsOverMatch (yes/no),

		• **<Sources>**: the UMLS source(s), • **<ConceptPIs>** • **<Status>**: 0/1/2 depending on if candidate is retained/excluded/pruned
<CandidateScore>	SU	The negative score of the candidate concept; the computation of this value is explained on pp. 5-9 of MetaMap Evaluation.
<CandidateCUI>	SU	The CUI of the candidate concept
<CandidateMatched>	SU	The candidate concept matched
<CandidatePreferred>	SU	The preferred name of the candidate concept
<MatchedWords Count="N"> <MatchedWord>	SR	The word(s) in the input text matched by the candidate
<SemTypes Count="N"> <SemType>	SR	The semantic type(s) of the candidate
<MatchMaps Count="N"> <MatchMap>	CR	A data structure representing • the correspondence of words in the candidate concept (**<TextMatchStart>** and **<TextMatchEnd>**) and words in the phrase (**<ConcMatchStart>** and **<ConcMatchEnd>**), and • the lexical variation (**<LexVariation>**) between the words in the candidate concept and the words in the phrase. For example, given the input text *obstructive sleep apnea* and the candidate concept *sleep apnea*, the matching words *sleep* and *apnea* are • the 2nd and 3rd words of the text, and • the 1st and 2nd words of the concept. There is no lexical variation, so the matchmap would therefore be **[[[2,3],[1,2],0]]**. For the candidate concept *sleep apneas*, the MatchMap would be the same, other than having lexical variation of 1 instead of 0.

<TextMatchStart>	SU	The position within the phrase words of the first matching word
<TextMatchEnd>	SU	The position within the phrase words of the last matching word
<ConcMatchStart>	SU	The position within the concept words of the first matching word
<ConcMatchEnd>	SU	The position within the concept words of the last matching word
<LexVariation>	SU	The degree of lexical variation between the words in the candidate concept and the words in the phrase; the computation of this value is explained on pp. 2-3 of MetaMap Evaluation.
<IsHead>	SU	Yes/no value denoting if the candidate concept includes the head of the phrase containing it
<IsOverMatch>	SU	Yes/no value denoting if the candidate concept is an overmatch, i.e., if it contains words on one or both ends that do not match the input text.
<Sources Count="N"> <Source>	SR	The UMLS vocabulary/ies in which the concept was found
<ConceptPIs Count="N"> <ConceptPI>	CR	The positional information of the concept, consisting of • **<StartPos>**: the 0-based character offset of the concept, counting from the beginning of the input text, and • **<Length>**: the character length of the string
<StartPos>	SU	The 0-based character offset of the string, counting from the beginning of the input text
<Length>	SU	The character length of the string
<Status>	SU	0, 1, or 2, representing if candidate was retained (0), excluded (1), or pruned (2)
<Mappings Count="N"> <Mapping>	CR	A set of candidate concepts making up the mapping for the phrase, consisting of • **<MappingScore>**: the negative score of the mapping, and • **<MappingCandidates>**: the candidate concept(s) participating in the

		mapping
<MappingScore>	SU	The negative score of the mapping; the computation of this value is explained on pp. 9-10 of MetaMap Evaluation.
<MappingCandidates Total="N"> <Candidate>	CU	The candidate concepts participating in a mapping