KAIROS Schema Data Format v2.3 (FINAL)

Last Modified: May 10th, 2023

The following are JSON-LD reserved keywords used by the KAIROS Schema Data Format (SDF). See JSON-LD v1.1 keyword documentation for details.

- "**`@context`**": Akin to an RDF prefix, this defines the short-hand names that are used throughout a JSON-LD document. It is REQUIRED in all file submissions. Program context should be included via a file reference. See the examples on GitHub or on Confluence.
- "**`@id`**": The id of an object that uniquely identifies it with an IRI. Essentially, this is how JSON-LD links to other object definitions: via their ids. It is REQUIRED in top-level documents, `instances`, `events`, `participants`, `entities`, and `relations` objects.
    - **IMPORTANT**: With the exception of document-level IDs, all IDs must follow the following naming convention:
        - `prefix:<KE_type>/<unique-5-digit-number>/<anything>`, where `KE_type` is one of *Events*, *Entities*, *Relations*, *Participants*, Values, and *Instances*, and `<anything>` is optional.
        - The scope of the unique numbers is within an entire `instance`.
    - IDs must not have whitespace or certain punctuation (%, \, ", ^, [,], {, }, |, <, >, and `), and cannot be `null`.
        - In addition, the following keywords are subject to the same punctuation rules: `entity`, `outlinks`, `predictionProvenance`, `provenanceID`, `wd_node`, `reference`, `relationObject`, `relationObject_prov`, `relationSubject`, `relationSubject_prov`, and `ta2entity`.
- "**`@container`**": Certain arrays are defined as **`@list`** containers, which means order is preserved.
- "**`@protected`**": With the exception of the `dwd` prefix, terms in the KAIROS context cannot be redefined/overridden.

The following are program-level JSON terms within the KAIROS SDF context, as defined by the KAIROS Schema Format (KSF) working group. These terms cannot be overridden by performer context. In many cases, a term may be applied equally to more than one type of element (e.g., an event and a participant).

- "**`sdfVersion`**": Defined at the document root level (sibling to `context` and `instances`), this is a REQUIRED string that identifies the target version of the Schema Data Format. The format of the version string is x.y.z where x, y, and z are numerical digits. Generally, the major version will change with each program phase, and the minor version will change with each quizlet/evaluation. The system will support specific versions of the SDF and is not expected to support every intermediate version.
- "**`ta2`**": Optionally defined at the document root level, setting this to `false` indicates that this is a TA1 schema library.
- "**`task2`**": REQUIRED at the document root level, setting this to `true` indicates that this is Task 2 output. This entails some validation changes, such as not requiring `confidence` (except at the instantiation level) and `provenanceData`.
- "**`ceID`**": This is defined and REQUIRED at the document root level. Set this to the complex event ID of the document set (Task 1) or human Graph G (Task 2).
    - For Task 1, the complex event ID is repeated in the `ce_id` field for each document in the document set as defined in the `parent_children.tab` file in the `docs` subfolder.
    - For Task 2, the complex event ID is specified in the `ceID` keyword of the human Graph G JSON-LD file.
- "**`version`**": a REQUIRED performer-specified version defined at the document level. Akin to the "filename" of the submission, it must, at a minimum, unambiguously indicate both TA1 and TA2 performers. It may also indicate algorithm versions or author in a succinct way.
- "**`instances`**": This is a REQUIRED keyword to organize separate complex event instantiations of a given TA1 schema library based on the document set specified by `ceID`. Multiple self-contained instantiations can be submitted in

accordance with submission instructions, but at most two instances may be returned per document set. Different libraries are instantiated in separate files.

- "**events**": An array of JSON objects (with an underlying graph structure), each of which contains a definition for an event node in the graph. It is defined and REQUIRED at the `instances` level. Any child event of an event must appear as an element in `events`.
- "**ta1ref**": A string that provides unambiguous traceability from a TA2 event or relation to its matching TA1 counterpart. It is REQUIRED in `events`, `instances`, and `relations` objects. If there is no matching TA1 element, such as an event from Graph G that is not present in the schema, then `ta1ref` must be set to "`none`".
    - Note that if `ta1ref` is set to "`none`", `provenance` must be supplied except for hierarchical event nodes.
    - `ta1ref` is composed of a unique 5-digit ID, prefixed with a two-letter code for KE type (e.g., SC/EV/RE for schema/event/relation, respectively). Everything after the seventh character is ignored.
- "**name**": Akin to `rdfs:label`, it is a *human-readable* English label or handle, typically only a word or two (e.g., "transport," "vaccinate," or "patient.") It is REQUIRED for `events`, `instances`, and `entities`, and recommended in `relations` objects to convey the meaning of the relation to humans.
    - Note that in `entities` and `events` objects, `name` is a handle for use by assessors, and should be the extracted text value or a meaningful human-readable handle (i.e. "alt-text") for non-text extractions.
    - If machine translation is used to populate `name`, then the original non-English text should be included in `origName`.
- "**description**": An English human-readable description, typically several words, a phrase, or even 1-2 sentences, but lacking spurious punctuation one might find in a file path or system-generated label. REQUIRED in `events`. If machine translation is used to populate `description`, then the original non-English text should be included in `origDescription`.
- "**origName**": If English names are created via machine translation, the original non-English version should be provided in the `origName` keyword for presentation in the UI, as analysts wish to see the original non-English source text for the purpose of keyword searches.
- "**origDescription**": If English descriptions are created via machine translation, the original non-English version should be provided in the `origDescription` keyword for presentation in the UI, as analysts wish to see the original non-English source text for the purpose of keyword searches.
- "**comment**": Akin to `rdfs:comment`, it provides human-readable detail. This is typically used in human-generated JSON-LD files, but is valid in machine-generated files. Can be a string or an array of strings.
- "**aka**": a synonym or string substitution for an event or entity. Specifying multiple synonyms is achieved by putting them in an array. It can be used in schema matching and propagated in TA2 output, but is not used in evaluation or by assessors.
- "**reference**": an external URI reference (or references, if placed in an array) to other datasets, knowledge bases, etc., e.g., VerbNet.
- "**privateData**": This keyword has been explicitly *removed* from the SDF, and will result in an invalid submission.
- "**temporal**": Defined in `events` objects, this object (or array of objects) stores event temporal values and related metadata. No one temporal keyword is required, but each temporal object must contain at least one REQUIRED temporal keyword (from those listed below), a REQUIRED `confidence` (Task 1), and optional `provenance`.
    - "**duration**": this is the duration of the event, expressed as an `xsd:duration`.
    - "**earliestStartTime**": this is the earliest possible start time of the event, expressed as an `xsd:dateTime`.
    - "**latestStartTime**": this is the latest possible start time of the event, expressed as an `xsd:dateTime`.
    - "**earliestEndTime**": this is the earliest end time of the event, expressed as an `xsd:dateTime`.
    - "**latestEndTime**": this is the latest end time of the event, expressed as an `xsd:dateTime`.
    - "**absoluteTime**": this is the absolute time of the event, expressed as an `xsd:dateTime`.
    - **NOTE**: All temporal data from human Graph G should be propagated in the TA2 output.

- "**confidence**": an xsd:float value between 0 and 1.0. In Task 1, it is REQUIRED in the events, values, instances, and temporal elements; in Task 2 it is only required at the instantiation level. A confidence of 1.0 is the highest possible value.
    - In events, it represents the probability that an instantiated event happened (or is predicted to happen), as informed by importance and optionality in the TA1 schema and confidence in the event matching from Graph G to the schema.
    - In instances, it represents the confidence that Graph G provides evidence of an instance of the TA1 schema.
    - In values, it indicates the confidence that an argument fills the specified participant role.
    - In temporal, it indicates the confidence in the correct timing/temporality of the event.
    - In Task 1, confidence in the information extraction should impact the confidence in the element.
    - The confidence of events and instances also determines assessment order of these elements.
    - The number of confidence values must match the corresponding number of provenance values (including relation provenance keywords, but excluding predictionProvenance). However, a single confidence value can be specified if all provenance values have the same confidence.
    - TA2 must leverage confidence to indicate what the multiple argument fillers indicate. For example:
        - "Tom, Dick, and Harry prepared the bomb": specify a confidence of 1.0 for all three arguments.
        - "Either Tom, Dick, or Harry prepared the bomb": specify confidence for all three alternate arguments reflecting how certain the system is that that argument fills the role—all values should be less than 1.0.
        - "Tom prepared the bomb along with Dick and/or Harry.": specify a confidence of 1.0 for Tom and less than 1.0 for Dick and Harry to specify uncertainty.
    - **NOTE**: In the case when it is clear that different arguments are connected to different other arguments, for example, "Miss Scarlet in the kitchen with the knife" and "Col. Mustard in the living room with the rope," then the event should be replicated in the output with the appropriate argument fillers. If supported by the source, one such event may outlink to another.
- "**modality**": Indicates additional properties for events, entity fillers (values), and relations, in accordance with LDC guidance. For events, valid values are generic, hedged, irrealis, and negated. For argument fillers, generic, negated and hedged are allowed. For relations, only negated and hedged are allowed. Multiple modality strings are defined as an array of elements.
- "**ta1explanation**": Allowed (but optional) in events, this string provided by TA1 gives an explanation/justification of the event—preferably grounded in the schema source. It can inform TA2, but is not propagated in their output, nor is it used by assessors.
- "**subgroup_events**": An unordered list of event @id references contained by an event in a parent-child relationship; must be consistent with each subgroup event's parent keyword. A given event @id cannot appear in more than one event's subgroup_events list. If the size of the list exceeds 7, consider creating additional levels of hierarchy.
- "**parent**": specifies the event's parent event in the hierarchy, if any. REQUIRED for all events, and must be consistent with parent's subgroup_events. Top-level events must specify a parent of "kairos:NULL".
- "**isTopLevel**": The root hierarchical event in a complex event instantiation must be flagged by setting the isTopLevel keyword to true.
- "**repeatable**": When set to true, indicates that the event can occur multiple times in the original TA1 schema. Multiple occurrences of repeatable events must be expanded by TA2 as distinct events in the instantiation, but the repeatable keyword should remain in the output whether or not the event is instantiated.
- "**outlinks**": One or more event @id references to *sibling* event nodes of the current event. It indicates the next event node(s) to be processed in the sequence. The event graph expressed by the collection of outlinks should be the transitive reduction of the event order graph. Outlink cycles will result in a rejected submission.

- "**children_gate**": In an event node, indicates the logical processing semantics of the node's children. REQUIRED when the subgroup_events keyword is specified. Expected values are "and", "or", and "xor" (case-insensitive), which have the following operational definitions:
    - **AND**: all children are expected to occur (e.g., "prep bomb").
    - **OR**: at least one child is expected to occur (e.g., "covid symptoms").
    - **XOR**: exactly one child is expected to occur (e.g., "manslaughter").

    NOTE: It is up to teams to decide how to use the schema for instantiation and prediction. The logical operators are meant to describe the prototypical case.
- "**wd_node**": A q-node (or p-node) from Wikidata. A q-node with QID Q1234 should be expressed as an @id in the form "wd:Q1234". A p-node with QID P1234 should be expressed as an @id in the form "wdt:P1234". The wd and wdt prefixes are defined in the KAIROS context file. It is REQUIRED in uninstantiated events and (TA1) entities linked to by the entity keyword. It is optional in participants.
    - To specify multiple wd_nodes, instead of "wd:Q1234 | wd:Q4321", create a list of wd_nodes: ["wd:Q1234", "wd:Q4321"], which has "OR" semantics. There is no way to specify AND wd_nodes.
    - The number of wd_labels and wd_descriptions must match the number of wd_nodes, but blank wd_labels are ignored.
    - NOTE: In the event that a DWD node does not have a Wikidata equivalent, you can refer to it via "dwd:Q1234". The dwd prefix is also be defined in KAIROS context, but can be overridden by performers.
- "**wd_label**": The Wikidata label of the wd_node defined at the same level (typically, an object in the entities, relations, or events array), REQUIRED when wd_node is present. Given that the label is set at a particular moment in time, it can become out of date as Wikidata evolves.
- "**ta2wd_node**": Found in instantiated ("real world") entities and events objects, this is a Wikidata q-node indicating the best q-node matching the extracted entity or event. It is REQUIRED in extra-relevant events and in entities linked to by the ta2entity keyword in values objects.
    - NOTE: See entry for wd_node regarding use of prefixes and how to specify multiple ta2wd_nodes.
    - If an event is used as an event argument, ta2wd_node should be used to specify the type of the event.
    - Extra-relevant events must specify a ta2wd_node instead of a wd_node, as there is no corresponding TA1 event.
- "**ta2wd_label**": Found in instantiated entities and events objects, and extra-relevant events, this is the label of the corresponding ta2wd_node. It is REQUIRED whenever ta2wd_node is present.
    - NOTE: See entry for wd_node for other details.
- "**wd_description / ta2wd_description**": This is the Wikidata description of the corresponding wd_node/ta2wd_node. It is REQUIRED in entities, events, participants, and relations when wd_node is provided. If the description is null or missing in Wikidata, then include an empty string.
- "**participants**": An ordered JSON array of participant arguments, REQUIRED in events when no subgroup_events are specified.
- "**roleName**": REQUIRED in participant objects. It is essentially the name/label of the slot, but connotes the semantic or linguistic role that is filled by the slot in the event. Guidance on how to specify role names can be obtained from the Cross-Program Ontology Group (XPO).
- "**entity**": REQUIRED in participant objects, this keyword links to the @id of an object in the entities array with a wd_node (not ta2wd_node), or—in the case of event arguments—a valid @id of an object in the events array. In this sense, entity can be thought of as a variable in the TA1 schema, which is filled by TA2 based on Graph G. In extra-relevant events or arguments, entity should be "kairos:NULL".
- "**values**": An object (or array of objects) associated with each event participant found in the source (documents or human Graph G). Each values object (or argument filler) contains:
    - An optional modality;
    - A REQUIRED @id, confidence (Task 1) and provenance; and

    - A REQUIRED `ta2entity` that specifies an `@id` of an object in the `entities` or `events` array. The `ta2entity` is the argument filler and must link to an object with a `ta2wd_node`.
    - NOTE: multiple argument fillers are modeled by multiple `values` objects, each with a unique `@id`. See additional guidance on multiple argument fillers in the description of the `confidence` keyword.
- "**`provenanceID`**": In Task 1, this is the unique string index of each object in the `provenanceData` table.
    - **IMPORTANT**: `provenanceID`s must follow the following naming convention:
        - `<prefix>:Provenance/<unique-5-digit-number>/<anything>`, where `<anything>` is optional.
- "**`provenance`**": In Task 1, this is a string index (or array of string indices) into the `provenanceData` table matching the `provenanceID` of one (or more) of the objects in the table. In Task 2, it is a reference to Graph G.
    - NOTE: TA2 should populate arguments in predicted events whenever possible and provide `provenance` based on the detected event(s) in which they participate.
    - **IMPORTANT**: If there is no provenance for a primitive event (e.g., a primitive schema step that wasn't found in the documents or human Graph G), TA2 must supply `predictionProvenance` instead.
- "**`predictionProvenance`**": REQUIRED in predicted *primitive* events. This is an unordered list of event, entity, or relation `@id`s that support the detection or prediction of an event. Please note:
    - Any event that appears in the schema but TA2 does not find evidence for is considered a *prediction*.
    - Hierarchical events (those with children as specified by `subgroup_events`) need not provide `predictionProvenance`.
    - All predictions must include `confidence` and `predictionProvenance`.
    - Predictions are evaluated in order of `confidence`, and only the first element of `predictionProvenance` is assessed.
- "**`entities`**": An array of objects, each of which defines an entity in the schema or a real-world entity from documents or human Graph G. Each object in the array has a REQUIRED `@id`, `name`, `wd_node`/`ta2wd_node`, `wd_label`/`ta2wd_label`, and `wd_description`/`ta2wd_description` for the entity. It can also specify a `reference` or `aka`.
- "**`relations`**": An array of objects that collectively specify entity-entity or event-event relations, ultimately establishing *subject* – *predicate* – *object* triples. Each object in the array is REQUIRED to contain an `@id`, `wd_node`, `relationObject`, `relationSubject`, `ta1ref`, `wd_label`, and `wd_description`. Relations can be defined in `events` objects or at the instantiation level. Relations defined at the top level could include containment or overlapping relations, equivalence relations between entities/events of different subevents in the hierarchy, or any other relation.
    - In TA2, instantiated relations are also REQUIRED to have `relationSubject_prov`, `relationObject_prov`, `relationProvenance`, `ta1ref`, and `confidence` (Task 1 only).
    - **NOTE**: for the best output, inverse relations should be detected and normalized. So given "Tom works for Company A" and "Company A employs Harry", the former relation should be converted to "Company A employs Tom".
    - **IMPORTANT**: NIST has requested that performers use a subset of Wikidata qnodes for temporal relations:
        - partial coincidence (Q65560376), and contains (P4330).
        - before (Q79030196) must not appear in `relation`s, but should instead be expressed as `outlinks`.
- "**`relationSubject`**": The single *subject* of a relation, typically a reference to an entity or event `@id`. It is REQUIRED in each `relations` array element.
- "**`relationObject`**": The *object* of a relation, typically a reference to an entity or event `@id`. It is REQUIRED in each `relations` array element. Relations with one subject but multiple objects should be separated into separate relations so that provenance can be shown for each individual relation.

- "**`relationSubject_prov`**": a URI index into the `provenanceData` table (Task1) or human Graph G (Task 2) showing provenance for the *subject* (Arg1) of an instantiated relation. It is REQUIRED in instantiated `relations` objects.
- "**`relationProvenance`**": a URI index into the `provenanceData` table (Task1) or human Graph G (Task 2) that demonstrates the presence of the relation in the documents. It is REQUIRED in instantiated `relations` objects.
- "**`relationObject_prov`**": a URI index into the `provenanceData` table (Task1) or human Graph G (Task 2) showing provenance for the *object* (Arg2) of an instantiated relation. It is REQUIRED in instantiated `relations` objects.
- "**`ta2entity`**": REQUIRED in `values` objects, this is the `@id` of an argument filler and links to an object in the `entities` or `events` array. That entity or event must have a `ta2wd_node`.
- "**`provenanceData`**": In Task 1, this is a REQUIRED table that contains details of extractions from a document that justify events, relations, and entities (slot fillers). All `provenance` specified in the output link to an object in `provenanceData` via their IDs. Each object contains the following elements:
    - A REQUIRED `provenanceID` string that serves as a matching ID for provenance specified elsewhere;
    - A REQUIRED `mediaType` of the extraction matching the media type of the specified `childID`;
    - A REQUIRED list of `parentIDs` that are parent documents to the specified `childID`.
    - A REQUIRED `childID` for the document; and
    - Additional REQUIRED fields based on the `mediaType` of the extraction.
    - An optional `sourceURL` that provides the URL of the parent document, for use in the Analyst UI.
    - **NOTE**: To specify audio provenance from a video source, create video provenance (not audio) and specify the entire bounding box. In general, video bounding boxes won't be used in assessment.
- "**`mediaType`**": the REQUIRED media type of a document extraction expressed as an IANA Media Type. The presence of the keyword is validated, and must match the media type of the corresponding `childID`.
- "**`childID`**": a REQUIRED LDC child_uid. It is the ID of the document/artifact and is validated against the current corpus of child_uids.
- "**`parentIDs`**": These REQUIRED LDC parent_uids are parent to the corresponding `childID`, and are validated against the current corpus of parent_uids. This can be an array because a single image/video/etc. can be embedded in more than one parent document.
- "**`sourceURL`**": This optional string provides the URL of the parent document as defined in the `parent_children.tab` file. This string is not validated, but will appear in a prospective UI for analysts, who may not trust systems without the ability to view the documents upon which the systems derive their output.
- "**`offset`**": an `xsd:integer` number of characters representing the offset into the LDC `rsd.txt` file. REQUIRED for text media types.
- "**`length`**": an `xsd:integer` number of characters in the span of a text extraction. REQUIRED for text media types.
- "**`boundingBox`**": an ordered four element array of `xsd:integer` dimensions (in pixels) corresponding to x1, y1, x2, and y2 coordinates (upperLeft-x, upperLeft-y, lowerRight-x, and lowerRight-y) from the bounding box of the image. REQUIRED for image and video media types.
- "**`keyframes`**": an array of LDC-supplied `xsd:integer` key frame IDs for key frame video extractions.
- "**`startTime`**": an `xsd:float` time stamp (in seconds) of the starting time of an audio recording or video mention. REQUIRED for audio and video media types.
- "**`endTime`**": an `xsd:float` time stamp (in seconds) of the ending time of an audio recording or video mention. REQUIRED for audio and video media types.

Alphabetical Keyword Quick Reference

Keyword	Location in document	Required?	Type / Notes
`@context`	document level	Yes	program context on GitHub
`@id`	document, entities, events, instances, participants, relations, values	Yes	Must follow a prescribed format, be a valid URI, and be unique
`absoluteTime`	temporal	No	`xsd:dateTime`
`aka`	events, entities	No	string or array of strings
`boundingBox`	provenanceData	Yes, in image & video media types; Task 1 only	ordered array of exactly four `xsd:integer`s
`ceID`	document	Yes	string
`childID`	provenanceData	Yes	string
`children_gate`	non-primitive events	Yes, if `subgroup_events` is present	`"and"`, `"or"`, or `"xor"`
`comment`	any object	No	string or array of strings
`confidence`	events, instances, relations, temporal, values	Yes, except relations; Task 1 only except instances	`xsd:float` between 0.0 and 1.0
`description`	events, instances	Yes, but in `events` only	string, human-readable English label
`duration`	temporal	No	`xsd:duration`
`earliestEndTime`	temporal	No	`xsd:dateTime`
`earliestStartTime`	temporal	No	`xsd:dateTime`
`endTime`	provenanceData	Yes, in audio & video media types; Task 1 only	`xsd:float`
`entities`	instances	Yes, if `events` is missing	object or array of objects
`entity`	participants	Yes	`@id` of entity or event
`events`	instances	Yes, if `entities` is missing	object or array of objects
`instances`	document	Yes	object or array of objects; self-contained instantiations
`isTopLevel`	events	No	indicates top of hierarchy
`keyframes`	provenanceData	No; Task 1 only	unordered array of `xsd:integer`s
`latestEndTime`	temporal	No	`xsd:dateTime`
`latestStartTime`	temporal	No	`xsd:dateTime`
`length`	provenanceData	Yes, in text media types; Task 1 only	`xsd:integer`

Keyword	Location in document	Required?	Type / Notes
`mediaType`	provenanceData	Yes	string IANA Media Type
`modality`	events, relations, values	No	string or array of strings subject to controlled vocab
`name`	entities, events, instances, relations	Yes, but merely encouraged for relations	string, human-readable English label
`offset`	provenanceData	Yes, in text media types; Task 1 only	`xsd:integer`
`origDescription`	events	No	use when using machine translation on descriptions
`origName`	entities, events, relations	No	use when using machine translation on names
`outlinks`	events	No	link(s) to event `@id`(s)
`parent`	events	Yes	link to event `@id`, or `"kairos:NULL"`
`parentIDs`	provenanceData	Yes	string or array of strings
`participants`	events	No	object or array of objects
`predictionProvenance`	events	Yes, in uninstantiated primitive events	array of event, entity, or relation `@id`s
`provenance`	events, temporal, values	Yes, in values and instantiated events	string ID(s) of object(s) in the `provenanceData` array, an ID from Graph G
`provenanceData`	document	Yes, in Task 1	object or array of objects
`provenanceID`	provenanceData	Yes	unique string ID for objects in the `provenanceData` array
`reference`	entities, events, participants, relations	No	`@id` or array of `@id`s of reference to external source(s) (not Wikidata)
`relationObject`	relations	Yes	`@id` of entity or event
`relationObject_prov`	relations	Yes, when `confidence` is supplied or `ta1ref` is `kairos:NULL`	string ID(s) of object(s) in the `provenanceData` array, an ID from Graph G
`relationProvenance`	relations	Yes, when `confidence` is supplied or `ta1ref` is `kairos:NULL`	string ID(s) of object(s) in the `provenanceData` array, an ID from Graph G
`relations`	events, instances	No	object or array of objects
`relationSubject`	relations	Yes	`@id` of entity or event

Keyword	Location in document	Required?	Type / Notes
relationSubject_prov	relations	Yes, when confidence is supplied or ta1ref is kairos:NULL	string ID(s) of object(s) in the provenanceData array, an ID from Graph G
repeatable	events	No	must expand as distinct events at instantiation time
roleName	participants	Yes	string, see XPO guidance
sdfVersion	document level	Yes	specially formatted string
sourceURL	provenanceData	No	Reference to the parent document, typically a URL
startTime	provenanceData	Yes, in audio & video media types; Task 1 only	xsd:float
subgroup_events	events	No	link(s) to event @id(s)
ta1explanation	events	No	string or array of strings
ta1ref	events, instances, relations	Yes	String ID of corresponding TA1 element, or "none"
ta2	document level	No	can omit in TA2 output
ta2entity	values	Yes	@id of extracted entity or event
ta2wd_description	entities, events	Yes, when ta2wd_node is present	string or array of strings, straight from Wikidata
ta2wd_label	entities, events	Yes, when ta2wd_node is present	string or array of strings, straight from Wikidata
ta2wd_node	entities, events	Yes, in extra-relevant events and when linked by ta2entity	@id or array of @ids, e.g., "wd:Q1234"
task2	document level	Yes	set to true for Task 2 output
temporal	events	No	object(s) with at least 1 of certain required keywords
values	participants	No	object or array of objects
version	document level	Yes	specially formatted string
wd_description	entities, events, participants, relations	Yes, when wd_node is present	string or array of strings, straight from Wikidata
wd_label	entities, events, participants, relations	Yes, when wd_node is present	string or array of strings, straight from Wikidata
wd_node	entities, events, participants, relations	Yes, in primitive events and when linked by entity	@id or array of @ids, e.g., "wd:Q1234"