



JSON User Guide v6.0

DCOI updates to FITARAMilestones.json due:
September 30, 2016.

Office of Management and Budget

General Guidelines:

-
- Step 1. **Work on your data in Notepad or a similar plain-text editor (not Microsoft Word). Microsoft Word will automatically replace numerous punctuation characters with formatted versions of the character. The presence of certain characters of this type (specifically, characters with non-UTF-8 encryption) will cause your data to be unreadable by OMB’s automated collection process.**

The following are examples of characters that cannot be processed: , , , , , , , , and . An arrow, for example, should instead look like this . Apostrophes and quotation marks should instead look like straight lines .

-
- Step 2. Use a JSON validator, such as the one linked on management.cio.gov/schema, to check that your agency’s file is a correctly-formatted, machine-readable JSON file. Simply copy and paste your drafted JSON file into the box on that page, and click on the “Validate” button. Errors or problems with the formatting of your file will appear below the box. You can then edit your file directly in the box to correct or address each error.

Once the file is clear, the site will show a green box like the one shown below:

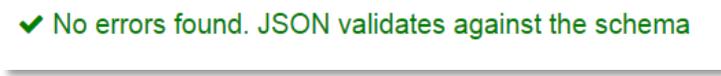


This indicates that your file is ready to be compared to the schema (see Step 3).

-
- Step 3. Use a JSON Schema Validator, such as the one linked on management.cio.gov/schema, to check your data. The left-hand side is for the “Schema” file found on management.cio.gov/schema. The right-hand-side box is for your agency’s data file. Errors in your file will immediately appear beneath the right-hand-side box.

NOTE: capitalization and space matter in JSON format. Make sure your entries to each field match EXACTLY to the options given in the corresponding table on management.cio.gov/schema.

Once all errors have been cleared, the site will display a message like the one below:



This indicates that your file is ready to be posted publicly on your digitalstrategy webpage.

-
- Step 4. Ensure that your file is located at the EXACT location specified by the policy requiring it (or that that precise URL redirects to your file). Again, capitalization matters.
-

I. About JSON:

JavaScript Object Notation (JSON) is a format for presenting data. Unlike data files that you might have worked with in Excel, JSON data is not organized as a table. Instead, JSON (similar to XML) is organized like the outline of a document: it has headings, sub-headings, sub-sub-headings, etc.

Also like the outline of a document, lower-level data is indented and punctuated in JSON in order to indicate that it falls within the most recent heading above it.

The “values” that you would normally put in a cell are instead placed immediately after the lowest level heading it relates to, and preceded by a colon.

EXAMPLE: A table of your agency’s Bureau CIOs might look like this in Microsoft Excel:

bureauCode	firstName	lastName	employmentType
12	Jane	Smith	GS

Your `bureaudirectory.json` file could contain the same information, but in JSON:

```
"bureauCode" : "12",  
"firstName" : "Jane",  
"lastName" : "Smith",  
"employmentType" : "GS"
```

How to Convert Data from a Table to JSON:

If you need to convert data stored in Excel or Sheets, and need to convert that data to JSON for the first time, management.cio.gov/schema contains a link to one of many tools available for free online which will do the bulk of the work for you. Prior to using one of these tools, however, consider taking the following steps:

1. You will want to first make sure all columns are typed **exactly** the same way as they are in the corresponding table describing the data on management.cio.gov/schema. Note that s p a c e s and capitalization must match **exactly** to the column names specified on management.cio.gov/schema, but the order of the columns need not be the same.
2. Also ensure that entries in each column conform to the rules for them as described in the table and instructions on management.cio.gov/schema. For example, if the table specifies that a date must be written as “MM/DD/YYYY”, then entries like “March 10, 2016”, “3-10-2016”, or “10/3/16” are *incorrect*. The correct entry is “03/10/2016”.
3. Ensure that text fields do not contain unusual or non-standard characters for punctuation. Note that the following are examples of characters that cannot be processed: , , , , , , , , and . An arrow, for example, should instead look like this . Apostrophes and quotation marks should instead look like straight lines . The difference between an ellipsis entered in Microsoft Word and three periods can be especially hard to detect. **The best way to avoid this at the start is to ensure that the data are only ever processed in Notepad or a similar simple-text editor, and to never use Microsoft Word with your JSON data files.**

II. How to Read your JSON File(s):

If you are unaccustomed to looking at non-tabular data structures, your file might initially be hard to read. It is important to understand that all of your data is still there, but it has been written out differently.

Instead of rows containing the different items in your file, the JSON file could be organized like a long list, where each row's information is presented in one spot, followed by a set of brackets and a comma, before the next "row"'s information.

The following might be the appearance of your data in Microsoft Excel:

milestoneID	milestoneDesc	milestoneTargetCompletionDate	milestoneStatus	milestoneStatusDesc	commonBaselineArea	dcoiArea
2	Conduct training for	2/1/2016	Complete	65 budget officer staff att	budgetFormulation	nonDataCenter

What you might be used to seeing as a "row", therefore, might look like this in JSON:¹

```
{
  "milestoneID": 2,
  "milestoneDesc": "Conduct training for department budget officers to share new budget formulation rules including CIO staff",
  "milestoneTargetCompletionDate": "2016/02/01",
  "milestoneStatus": "Complete",
  "milestoneStatusDesc": "65 budget officer staff attended the training, including representatives from 12 agencies.",
  "commonBaselineArea": "budgetFormulation",
  "dcoiArea": "nonDataCenter"
},
```

Note that the above information could alternatively be structured like the below, which more closely resembles the original row with columns:²

```
{
  "milestoneID": 2, "milestoneTargetCompletionDate": "2016/02/01", "milestoneStatus": "Complete", "commonBaselineArea": "budgetFormulation", "dcoiArea": "nonDataCenter"
},
```

Much like how you would write out a list of items, the last entry (row) in your JSON data will *not* be followed by a comma, but all previous items must be separated by commas. Similarly, within your data, each field (comparable to "columns") will have to be separated by commas except for after the last one.

Also note that curly brackets ({ or }) are used to contain groups of things, such as groups of information in each row. This indicates that all of the information contained within the brackets describes one thing.

Rows like the ones above can be added in a list, one after the other. At any time, another "row" of data may be added to your data file in the same way. See the **Appendix** of this document for empty, template entries for each of OMB's data collections on management.cio.gov/schema. You may simply copy and paste this into your file and replace the text contained therein with your data, to add another item to your data file.

¹ Screenshot contains one milestone from the FITARAmilestones.json demo file. The full demo file can be found on management.cio.gov/schema.

² Note the text fields "milestoneDesc" and "milestoneStatusDesc" have been omitted from the second screenshot for space.

III. How to Ensure Your Data is in JSON File Format:

Use a JSON validator, such as the one linked on management.cio.gov/schema (or another of your choosing), to check that your agency's file is a correctly-formatted, machine-readable JSON file. Simply copy and paste your drafted data file into the box on that page, and click on the "Validate" button.

Errors or problems with the formatting of your file will appear below the box. You can then edit your file directly in the box to correct or address each error. Once the file is clear, the site will show a green box like the one shown below (shown for the site linked on management.cio.gov/schema):



IV. About JSON Schemas:

A data “schema” is just a file that describes any requirements around formatting, type, or content that are needed to be compatible with another file. It describes how the data should be typed and presented to be processed accurately.

Because OMB collects JSON data files from each agency, each agency’s file must have a shared understanding of how the data are to be presented. For example, if one field asked agencies to report spending, all agencies must be using the same units (e.g. the difference between answering “1” when the units are dollars and when the units are millions of dollars, is pretty big).

Numerous tools available online can compare a data file to a schema file. OMB recommends one on management.cio.gov/schema that is highly user-friendly, but there are numerous more complex tools available. These tools will require you to copy and paste the schema file for the data you are working on, plus your own data file you wish to post publicly, and can compare them to locate and show you any errors found in your data file compared to the expectations for the data OMB specifies in the schema.

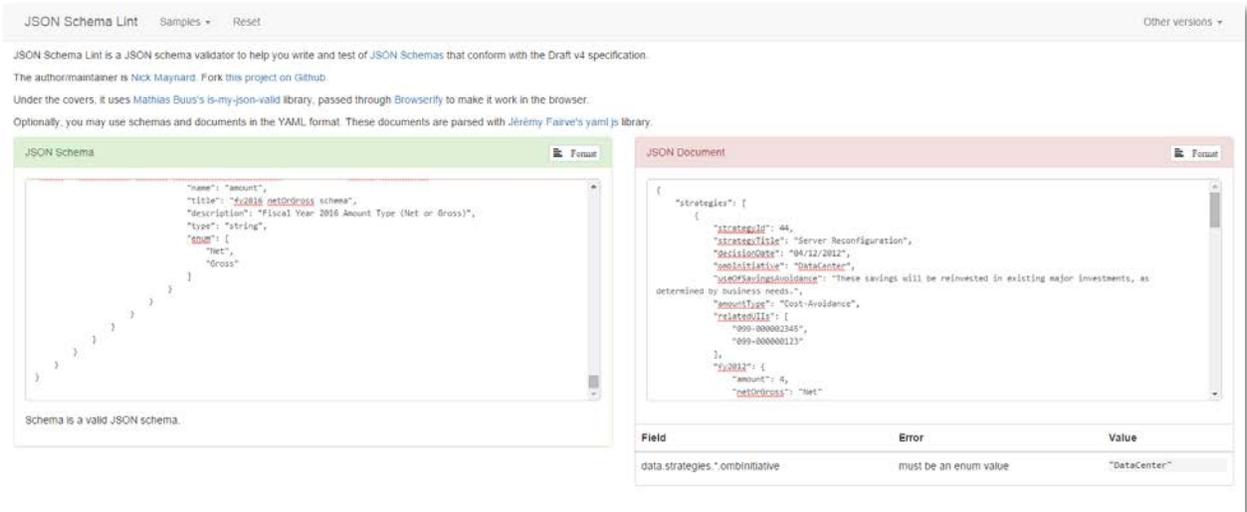
EXAMPLE: If a field asks agencies to report spending on IT in millions of dollars, the schema written by OMB might say something like, “this field should only have numbers and one decimal point in it” (e.g. \$10.5 million dollars should be entered as “10.50” or “10.5”). If the data file you wish to submit contains “\$10.5” in this spending field, this would be flagged as a schema error because the entry “\$10.5” contains a character that the schema does not allow (the dollar sign).

V. How to Use a JSON Schema:

Numerous tools available online can compare a data file to a schema file. OMB recommends one on management.cio.gov/schema that is highly user-friendly, but there are numerous more complex tools available.

To get started, navigate to management.cio.gov/schema and locate the schema file for your data file. Click on the link to the schema to view it in your browser. Press Ctrl+A on your keyboard to select the entire schema, and then copy it (Ctrl + C).

In a new webpage tab, navigate to the JSON Schema Validator linked on management.cio.gov/schema (or another schema validator of your choice). Notice that the webpage has two white boxes in it. Paste (Ctrl + V) the schema into the box on the left, then copy and paste your JSON File in the box on the right. Errors in your file will display immediately below the right-hand-side box, which will be bordered in pink until all errors are cleared. The screenshot below shows what this screen might look like using the JSON Schema Validator linked on management.cio.gov/schema and the costsavings.json schema and demo files:



NOTE: capitalization and spaces matter in JSON format. Make sure your entries to each field in your JSON file match **EXACTLY** to the options given in the appropriate table for your data collection on management.cio.gov/schema.

VI. How to Post Your Validated JSON File:

Once the process described in the preceding sections of this document has been completed and your JSON file is valid and free of schema errors, the last step is to upload your file to the location specified by the policy requiring the data collection. This might require you to submit a ticket with your agency's IT or Web Administrator office. Note that capitalization matters in the URL of your JSON file. Redirecting to the file (from the precise URL specified by the policy) is acceptable, as OMB's automated process for pulling JSON files can follow the redirect.

The following are the required locations for agencies' JSON files:

Agency FITARA Milestones	[agencyhomepage].gov/digitalstrategy/FITARAMilestones.json
DCOI Strategic Plans *	[agencyhomepage].gov/digitalstrategy/datacenteroptimizationsstrategicplan.json * Note: A separate, signed document is also required to be submitted, in addition to posting your JSON file. See OMB M-16-19 .
Bureau IT Leadership Directory	[agencyhomepage].gov/digitalstrategy/bureaudirectory.json
CIO Governance Board Membership List	[agencyhomepage].gov/digitalstrategy/governanceboards.json
IDC Realized Cost Savings and Avoidance	[agencyhomepage].gov/digitalstrategy/costsavings.json

APPENDIX

Adding a Milestone to your FITARA Milestones:

Copy and paste the following into your file, in between two existing milestones:

```
{
  "milestoneID": #,
  "milestoneDesc": "Abc",
  "milestoneTargetCompletionDate": "YYYY/MM/DD",
  "milestoneStatus": "Abc",
  "milestoneStatusDesc": "Abc.",
  "commonBaselineArea": "Abc",
  "dcoiArea": "Abc"
},
```

Then, replace the placeholder text with the appropriate text from your data.

Adding a Leader to your Bureau IT Leadership Directory:

Copy and paste the following into your file, in between two existing leaders:

```
{
  "bureauCode" : "##",
  "firstName" : "Abc",
  "lastName" : "Abc",
  "employmentType" : "Abc",
  "typeOfAppointment" : "Abc",
  "otherResponsibilities": "Optional abc.",
  "evaluationRatingOfficialTitle" : "Abc",
  "evaluationReviewingOfficialTitle" : "Optional abc.",
  "keyBureauCIO" : "Yes/No"
},
```

Then, replace the placeholder text with the appropriate text from your data.

Adding a Governance Board to your CIO Governance Board Membership List:

Copy and paste the following into your file, in between two existing governance boards:

```
{
  "governanceBoardName" : "Abc",
  "programCodeFPI" : "###-###",
  "bureauCode" : "##",
  "cioInvolvementDescription" : "Optional abc."
},
```

Then, replace the placeholder text with the appropriate text from your data.

Adding a Strategy to your IDC Realized Cost Savings and Avoidance:

Copy and paste the following into your file, in between two existing strategies:

```
{
  "strategyId": #,
  "strategyTitle": "Abc",
  "decisionDate": "MM/DD/YYYY",
  "ombInitiative": "Abc",
  "useOfSavingsAvoidance": "Optional abc."
  "amountType": "Abc",
  "relatedUIIs": [
    "###-#####"
  ],
  "fy2012": {
    "amount": #,
    "netOrGross": "Abc"
  },
  "fy2013": {
    "amount": #.##,
    "netOrGross": "Abc"
  },
  "fy2014": {
    "amount": #,
    "netOrGross": "Abc"
  },
  "fy2015": {
    "amount": #,
    "netOrGross": "Abc"
  },
  "fy2016": {
    "amount": #,
    "netOrGross": "Abc"
  }
},
```

Then, replace the placeholder text with the appropriate text from your data.