

Internet Engineering Task Force  
Internet-Draft  
Intended status: Informational  
Expires: December 11, 2011

P. Bryan, Ed.  
ForgeRock US, Inc.  
June 9, 2011

A JSON Media Type for Describing Partial Modifications to JSON Documents  
[draft-pbryan-json-patch-01](#)

Abstract

JSON (JavaScript Object Notation) Patch defines the media type "application/patch+json", a JSON-based document structure for specifying partial modifications to apply to a JSON document.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 11, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Internet-Draft

JSON Patch Media Type

June 2011

## Table of Contents

|                             |                                       |                   |
|-----------------------------|---------------------------------------|-------------------|
| <a href="#">1.</a>          | Introduction . . . . .                | <a href="#">3</a> |
| <a href="#">2.</a>          | Conventions . . . . .                 | <a href="#">3</a> |
| <a href="#">3.</a>          | Motivations . . . . .                 | <a href="#">3</a> |
| <a href="#">4.</a>          | Patch Object Structure . . . . .      | <a href="#">3</a> |
| <a href="#">5.</a>          | Operations . . . . .                  | <a href="#">3</a> |
| <a href="#">5.1.</a>        | add . . . . .                         | <a href="#">4</a> |
| <a href="#">5.2.</a>        | remove . . . . .                      | <a href="#">4</a> |
| <a href="#">5.3.</a>        | replace . . . . .                     | <a href="#">4</a> |
| <a href="#">6.</a>          | Error Handling . . . . .              | <a href="#">4</a> |
| <a href="#">7.</a>          | Acknowledgements . . . . .            | <a href="#">4</a> |
| <a href="#">8.</a>          | IANA Considerations . . . . .         | <a href="#">5</a> |
| <a href="#">9.</a>          | Security Considerations . . . . .     | <a href="#">5</a> |
| <a href="#">10.</a>         | References . . . . .                  | <a href="#">5</a> |
| <a href="#">10.1.</a>       | Normative References . . . . .        | <a href="#">5</a> |
| <a href="#">10.2.</a>       | Informative References . . . . .      | <a href="#">5</a> |
| <a href="#">Appendix A.</a> | Examples . . . . .                    | <a href="#">5</a> |
| <a href="#">A.1.</a>        | Adding an Object Property . . . . .   | <a href="#">6</a> |
| <a href="#">A.2.</a>        | Adding an Array Element . . . . .     | <a href="#">6</a> |
| <a href="#">A.3.</a>        | Removing an Object Property . . . . . | <a href="#">6</a> |
| <a href="#">A.4.</a>        | Removing an Array Element . . . . .   | <a href="#">7</a> |
| <a href="#">A.5.</a>        | Replacing a Value . . . . .           | <a href="#">7</a> |
| <a href="#">Appendix B.</a> | Changelog . . . . .                   | <a href="#">8</a> |
| <a href="#">Appendix C.</a> | Open Issues . . . . .                 | <a href="#">8</a> |
|                             | Author's Address . . . . .            | <a href="#">8</a> |

## 1. Introduction

This Internet-Draft proposes the JSON-Patch media type "application/patch+json", a JSON structure for specifying partial modifications to apply to a JSON document.

## 2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

## 3. Motivations

JavaScript Object Notation (JSON) [[RFC4627](#)] is a popular format for the storage and transmission of structured data. As a result, numerous web service APIs support the JSON media type.

The HTTP PATCH [[RFC5789](#)] specification extends HTTP with a new method to perform partial modifications to resources. A JSON-based patch document type is required to modify JSON documents using this method.

## 4. Patch Object Structure

A JSON Patch document is a JSON document that contains an array of objects. Each object contains a patch operation to apply to a target JSON document.

A sample JSON Patch document:

```
[
  { "remove": "/a/b/c" },
  { "add": "/a/b/c", "value": "foo" },
```

```
{ "replace": "/a/b/c", "value": "bar" }
]
```

Patch operations are applied sequentially in the order they appear in the array. After a patch operation has been applied, the resulting JSON document becomes the target for the next patch operation. This process repeats until all patches are successfully applied.

## [5.](#) Operations

Operations are specified in the name of a property in the patch

Bryan

Expires December 11, 2011

[Page 3]

---

Internet-Draft

JSON Patch Media Type

June 2011

operation object, with that property's value being a string containing the [JSON Pointer] that references the node to be affected.

### [5.1.](#) add

Adds a new value into an object or array. The value to be added to the object or array is specified in the "value" property.

If adding to an object, it is an error condition if the property to be added in the object already exists.

If adding to an array, any elements at or above the specified index are shifted one position to the right (i.e. indexes are incremented). It is an error condition if the addition would result in sparse allocation of any array elements.

### [5.2.](#) remove

Removes the value specified by the JSON Pointer value.

If removing an element from an array, any elements above the specified index are shifted one position to the left (i.e. indexes are decremented).

It is an error condition if the value to be removed does not exist.

### [5.3.](#) replace

Replaces the value referenced by the JSON Pointer with a new value. Syntactically equivalent to performing a "remove" operation, followed immediately by an "add" operation. The value to be replaced is specified in the "value" property.

It is an error condition if the value to be replaced does not exist.

## 6. Error Handling

In the event of an error condition, evaluation of the JSON Patch document ceases. Enforcing safety and/or idempotency of a JSON Patch document is specified by and enforced by the implementation.

## 7. Acknowledgements

The structure of a JSON Patch document was informed by the XML Patch document [[RFC5261](#)] specification.

Bryan

Expires December 11, 2011

[Page 4]

---

Internet-Draft

JSON Patch Media Type

June 2011

## 8. IANA Considerations

The proposed MIME media type for the JSON Patch document is "application/patch+json".

Type name: application

Subtype name: patch+json

Required parameters: none

Optional parameters: none

## 9. Security Considerations

This specification is based on the JSON format, therefore has the same security considerations as [[RFC4627](#)].

## 10. References

## 10.1. Normative References

[JSON Pointer]

Bryan, P. and K. Zyp, "JSON Pointer", June 2011, <<http://tools.ietf.org/html/draft-pbryan-zyp-json-pointer-00>>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

[RFC4627] Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", [RFC 4627](#), July 2006.

## 10.2. Informative References

[RFC5261] Urpalainen, J., "An Extensible Markup Language (XML) Patch Operations Framework Utilizing XML Path Language (XPath) Selectors", [RFC 5261](#), September 2008.

[RFC5789] Dusseault, L. and J. Snell, "PATCH Method for HTTP", [RFC 5789](#), March 2010.

## Appendix A. Examples

Bryan

Expires December 11, 2011

[Page 5]

---

Internet-Draft

JSON Patch Media Type

June 2011

### A.1. Adding an Object Property

An example target JSON document:

```
{
  "foo": "bar"
}
```

A JSON Patch document:

```
[
  { "add": "/baz", "value": "qux" }
]
```

The resulting JSON document:

```
{
  "baz": "qux",
  "foo": "bar"
}
```

### [A.2.](#) Adding an Array Element

An example target JSON document:

```
{
  "foo": [ "bar", "baz" ]
}
```

A JSON Patch document:

```
[
  { "add": "/foo/1", "value": "qux" }
]
```

The resulting JSON document:

```
{
  "foo": [ "bar", "qux", "baz" ]
}
```

### [A.3.](#) Removing an Object Property

An example target JSON document:

```
{
  "baz": "qux",
  "foo": "bar"
}
```

```
}
```

A JSON Patch document:

```
[
  { "remove": "/baz" }
]
```

The resulting JSON document:

```
{
  "foo": "bar"
}
```

#### [A.4.](#) Removing an Array Element

An example target JSON document:

```
{
  "foo": [ "bar", "qux", "baz" ]
}
```

A JSON Patch document:

```
[
  { "remove": "/foo/1" }
]
```

The resulting JSON document:

```
{
  "foo": ["bar", "baz"]
}
```

#### [A.5.](#) Replacing a Value

An example target JSON document:

```
{
  "baz": "qux",
  "foo": "bar"
}
```

A JSON Patch document:

```
[
  { "replace": "/baz", "value": "boo" }
]
```



The resulting JSON document:

```
{
  "baz": "boo",
  "foo": "bar"
}
```

## [Appendix B.](#) Changelog

Draft 00

- o Initial draft.

Draft 01

- o Switched from JSONPath to JSON Pointer to reference nodes.
- o Removed "op"; property name now contains operation to perform.
- o Removed "element"; JSON Pointer now contains target node.
- o Added "replace" operation to decrease operation verbosity.

## [Appendix C.](#) Open Issues

Interest has been expressed in conforming to a uniform PUT/DELETE interface, and providing for idempotent and non-idempotent operations within the patch document format.

### Author's Address

Paul C. Bryan (editor)  
ForgeRock US, Inc.  
201 NE Park Plaza Drive Suite 196  
Vancouver, WA 98684  
USA

Phone: +1 604 783 1481  
Email: paul.bryan@forgerock.com