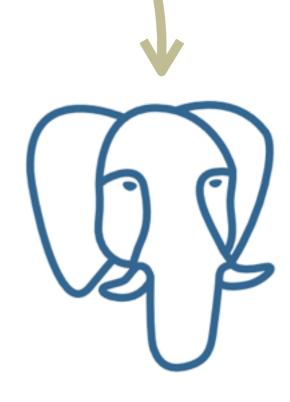


### Hello, I'm Daniel Gustafsson

### Pivotal







commit 8c1de5fb0010ae712568f1706b737270c3609bd8

Author: Peter Eisentraut <peter\_e@gmx.net>

Date: Thu Dec 21 16:05:16 2006 +0000

Initial **SQL/XML** support: xml data type and initial set of functions.



commit 5384a73f98d9829725186a7b65bf4f8adb3cfaf1

Author: Robert Haas <rhaas@postgresql.org>

Date: Tue Jan 31 11:48:23 **2012** -0500

Built-in JSON data type.

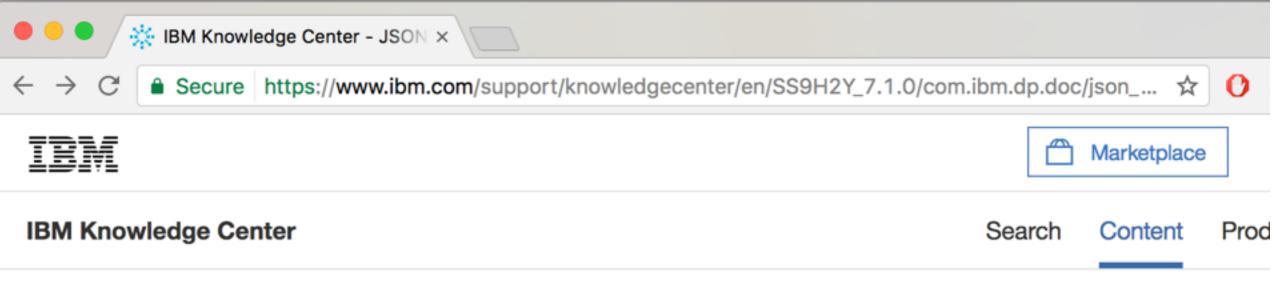


### 

# 1 + 1







DataPower Gateways > DataPower Gateways 7.1.0 > Supplemental >

JSONx

JSON and JSONx >

Version ... ∨



JSONx is an IBM® standard format to represent JSON as XML. The appliance converts JSON messages that are specified as JSON message type to JSONx. The appliance provides a style sheet that you can use to convert JSONx to JSON.

JSONx conversion rules specify how a DataPower® service converts a JSON structure to JSONx (XML).

The DataPower appliance provides the default jsonx2json.xsl style sheet for converting JSONx to JSON. This style sheet is in the store: directory.

```
"name": "John Smith",
"address": {
  "streetAddress": "21 2nd Street",
  "city": "New York",
  "state": "NY",
  "postalCode": 10021,
"phoneNumbers": [
  "212 555-1111",
  "212 555-2222"
"additionalInfo": null,
"remote": false,
"height": 62.4,
"ficoScore": " > 640"
```

```
<?xml version="1.0" encoding="UTF-8"?>
<json:object xsi:schemaLocation="http://www.datapower.com/schemas/</pre>
json jsonx.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:json="http://www.ibm.com/xmlns/prod/2009/jsonx">
  <json:string name="name">John Smith</json:string>
  <json:object name="address">
    <json:string name="streetAddress">21 2nd Street</json:string>
    <json:string name="city">New York</json:string>
    <json:string name="state">NY</json:string>
    <json:number name="postalCode">10021</json:number>
 </json:object>
 <json:array name="phoneNumbers">
    <json:string>212 555-1111</json:string>
    <json:string>212 555-2222</json:string>
 </json:array>
  <json:null name="additionalInfo" />
  <json:boolean name="remote">false</json:boolean>
  <json:number name="height">62.4</json:number>
  <json:string name="ficoScore"> > 640</json:string>
</json:object>
```

Network Working Group
Internet-Draft
Expires: November 3, 2011

B. MuschettR. SalzM. SchenkerIBMMay 2, 2011

JSONx, an XML Encoding for JSON draft-rsalz-jsonx-00.txt

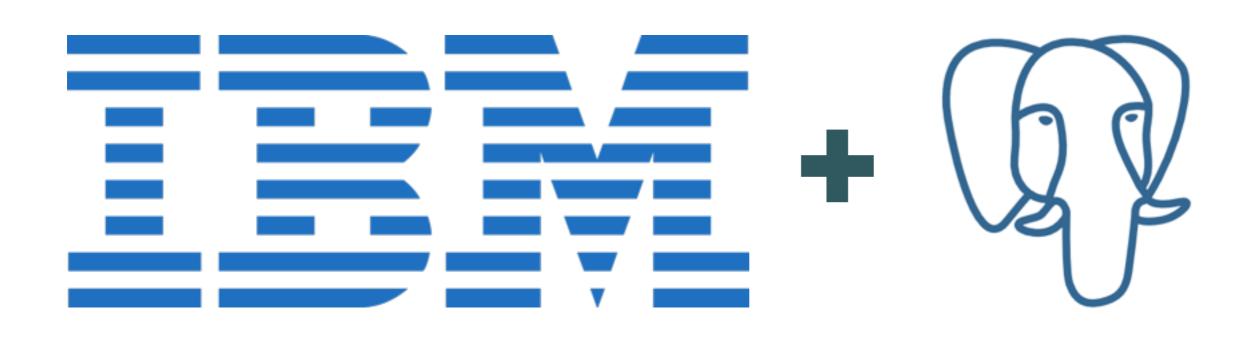
### Abstract

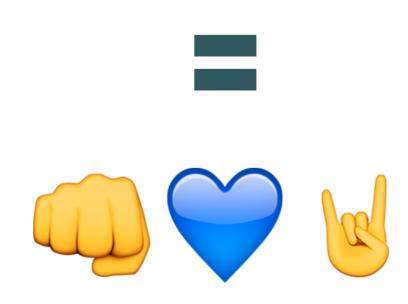
This document specifies a mapping between JSON (RFC 4627) and XML. The mapping maintains a high degree of fidelity. It is used by several IBM products.

### 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/.





```
postgres=# select jsonx_build_object('{"zip": 11249,
"phone": ["1-800-AWESOME", "1-800-TURKEY"]}');
             jsonx_build_object
<json:number name="zip">11249</json:number>+
<json:array name="phone">
   <json:string>1-800-AWESOME</json:string>
   <json:string>1-800-TURKEY</json:string>
</json:array>
(1 row)
```

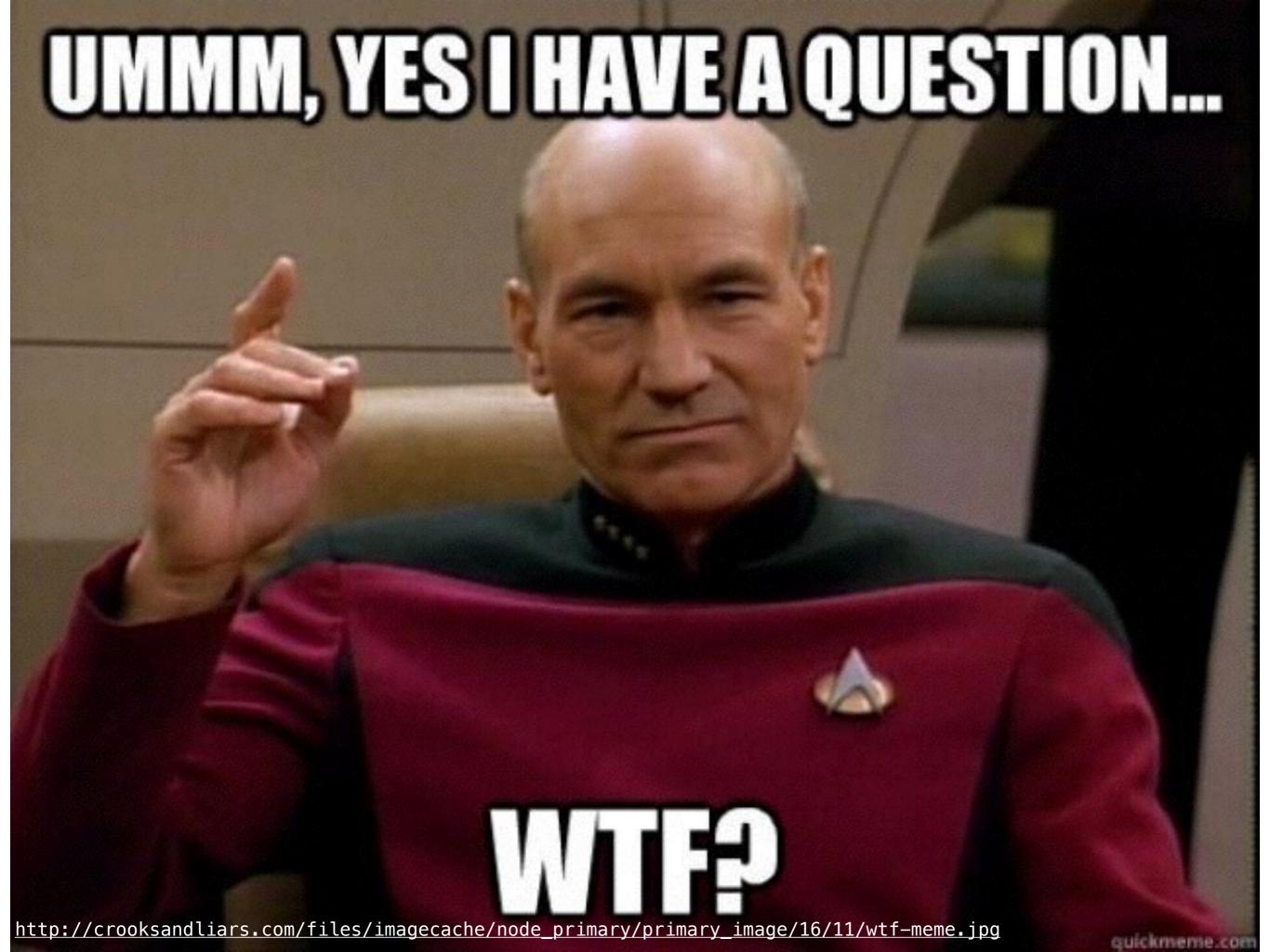
```
["algol", "fortran"]}');
                       jsonx_build_document
<json:object xsi:schemaLocation="http://www.datapower.com/schemas/</pre>
json jsonx.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:json="http://www.ibm.com/xmlns/prod/2009/jsonx">
   <json:array name="languages">
     <json:string>algol</json:string>
     <json:string>fortran</json:string>
   </json:array>
</json:object>
(1 row)
```

postgres=# select jsonx\_build\_document('{"languages":

```
postgres=# select xmlexists('//*[@name]' PASSING
jsonx_build_document('{"zip": 11249, "phone":["1-800-
AWESOME","1-800-TURKEY"]}'));

xmlexists
______
t
```

(1 row)



### In all seriousness

commit a570c98d7fa0841f17bbf51d62d02d9e493c7fcc Author: Andrew Dunstan <andrew@dunslane.net> Date: Fri Mar 29 14:12:13 2013 -0400

Add new JSON processing functions and parser API.

## pg\_parse\_json();

```
JsonLexContext
                *lex;
JsonxState
                *state;
JsonSemAction
                *sem;
text *json = cstring to text(json text);
sem = palloc0(sizeof(JsonSemAction));
state = palloc(sizeof(JsonxState));
sem->semstate = (void *) state;
lex = makeJsonLexContext(json, true);
pg_parse_json(lex, sem);
```

```
JsonLexContext
                *lex;
JsonxState
                *state;
JsonSemAction
                *sem;
text *json = cstring to text(json text);
sem = palloc0(sizeof(JsonSemAction));
state = palloc(sizeof(JsonxState));
sem->semstate = (void *) state;
sem->object_field_start = object_field_start;
sem->scalar = get_scalar;
lex = makeJsonLexContext(json, true);
pg_parse_json(lex, sem);
```

```
typedef struct JsonxState
{
  char *key;
  char *value;
} JsonxState;
```

```
static void
get scalar(void *state, char *token,
           JsonTokenType type)
 JsonxState *s = (JsonxState *) state;
 switch(type)
   case JSON_TOKEN_STRING:
     s->value = pstrdup(token);
     break;
   case JSON TOKEN NUMBER:
     break;
   default:
     break;
```

```
$ git diff --staged --stat
src/backend/utils/adt/Makefile | 2 +-
src/backend/utils/adt/jsonx.c | 198 ++++++++++
src/backend/utils/adt/xml.c | 2 +-
src/include/catalog/pg_proc.h | 3 +
4 files changed, 203 insertions(+), 2 deletions(-)
```

2011. Posconfield

### Thanks / I'm sorry

daniel@yesql.se @d\_gustafsson