Node.js, DB2, RPG

talking at last

Aaron Bartell abartell@krengeltech.com

LITMIS.COM - Open Source for i

Agenda

- Why should I care about Node.js?
- Javascript for noobs
- DB2 Access for i
- Toolkit for i (aka iToolkit)

What is Node.js? Make me care.

Server-side Javascript, uses Google's V8

Biggest benefit: One language for both client *(browser)* and server *(IBM i)*.

BIG TIME SAVER!!

Ported and supported by IBM.

Home: http://bit.ly/nodejs_ibmi

Non-blocking I/O. Not unique to Node.js, though they made it popular.

w3schools.com/js - Javascript, getting started
developers.google.com/v8 - Google V8
en.wikipedia.org/wiki/MIT_License - MIT license

A Little Javascript for noobs

The `require` feature in Node.js is similar to RPG's `/copy` - bring in outside functionality. See below "**Answers**" link because this is a busy topic.

```
1.  var ibmi = require('/home/aaron/git/nodejs_playing/ibmi');
2.  var user = ibmi.current_user();
```

The `exports` syntax declares the functions to make available, like `export` on an RPG sub procedure.

ibmi.js

```
1.  function current_user(){
2.    return rtvjoba('USER');
3.  }
4.
5.  exports.current_user = current_user;
```

http://stackoverflow.com/questions/16521471/relation-between-commonjs-amd-and-requirejs - Answers

noobs continued...

Javascript **callbacks** are like RPG **procedure pointers...**

```
1.  function my_callback(data) {
2.    console.log('data: ' + data);
3.  }
4.
5.  function hi(callback) {
6.    callback('get it?');
7.  }
8.
9.  hi(my_callback);
```

callbackhell.com - Learn how to write Javascript callbacks
javascriptissexy.com/understand-javascript-callback-functions-and-use-them - Callback functions

noobs continued...

Anonymous function assigned to **variable** *(mind blown)*...

```
1.  var my_callback = function(data) {
2.    console.log('data: ' + data);
3.  };
4.
5.  function hi(callback) {
6.    callback('get it?');
7.  }
8.
9.  hi(my_callback);
```

w3schools.com/js/js_function_definition.asp - Anonymous functions

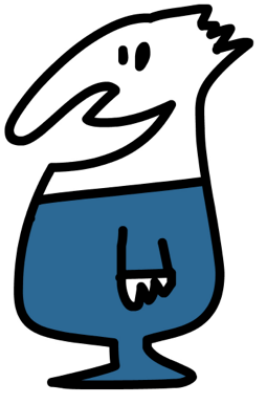
noobs continued...

Inline anonymous function...

```
1.  function hi(callback) {
2.    callback('get it?');
3.  }
4.
5.  hi(function(data) {
6.    console.log('data: ' + data);
7.  });
```

w3schools.com/js/js_function_definition.asp - Anonymous functions

Talking to IBM i



DB2 Access for i

- **Descr:** API set for manipulating DB2 for i from Node.js
- **Docs:** http://bit.ly/nodejs_db2foriaccess
- **IFS Location:** /QOpenSys/QIBM/ProdData/Node/os400/db2i/lib/db2

Toolkit for i

- **Descr:** API set for interfacing with XMLSERVICE.
- **Docs:** http://bit.ly/nodejs_toolkitfori
- **IFS Location:**
  /QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit

Config IBM i Environment

Do yourself a favor and DON'T use `CALL QP2TERM`, use SSH client instead.

Run the following in a PASE shell:

```
$ export PATH=/QOpenSys/QIBM/ProdData/Node/bin:$PATH
$ export LIBPATH=/QOpenSys/QIBM/ProdData/Node/bin:$LIBPATH
```

Now you can invoke `node` binary from anywhere in the IFS.

```
$ node -v
v0.10.29
```

Put the above exports into `node_env.sh` and then easily invoke it *(note the period!)*

```
$ . node_env.sh
```

bitbucket.org/litmis/nodejs/wiki/environment - IBM i Node.js Environment config
bit.ly/chrome_ssh - Secure Shell in Chrome *(browser)*

What are YiPs?

YiPs = Young i Professionals

- Began as a group within COMMON USA as a way for young people to find each other at the COMMON USA annual conference.
- **Goals:** educate, advocate, and get involved!
- IBM uses YiPs to share open source that originates from within their walls *(i.e. xmlservice, Ruby gems, PHP code, many tutorials on Git, PASE, ssh, FastCGI, etc)*
- **"Join"** YiPs via LinkedIn YiPs group: **linkd.in/1iG3olV**
- **Only need to be young at heart! :-)**

YoungiProfessionals.com - Home site
linkd.in/1iG3olV - LinkedIn YiPs group

What is XMLSERVICE?

… a single library of Open Source RPG code that enables XML scripting calls of IBM i resources using most any language available on your platform.

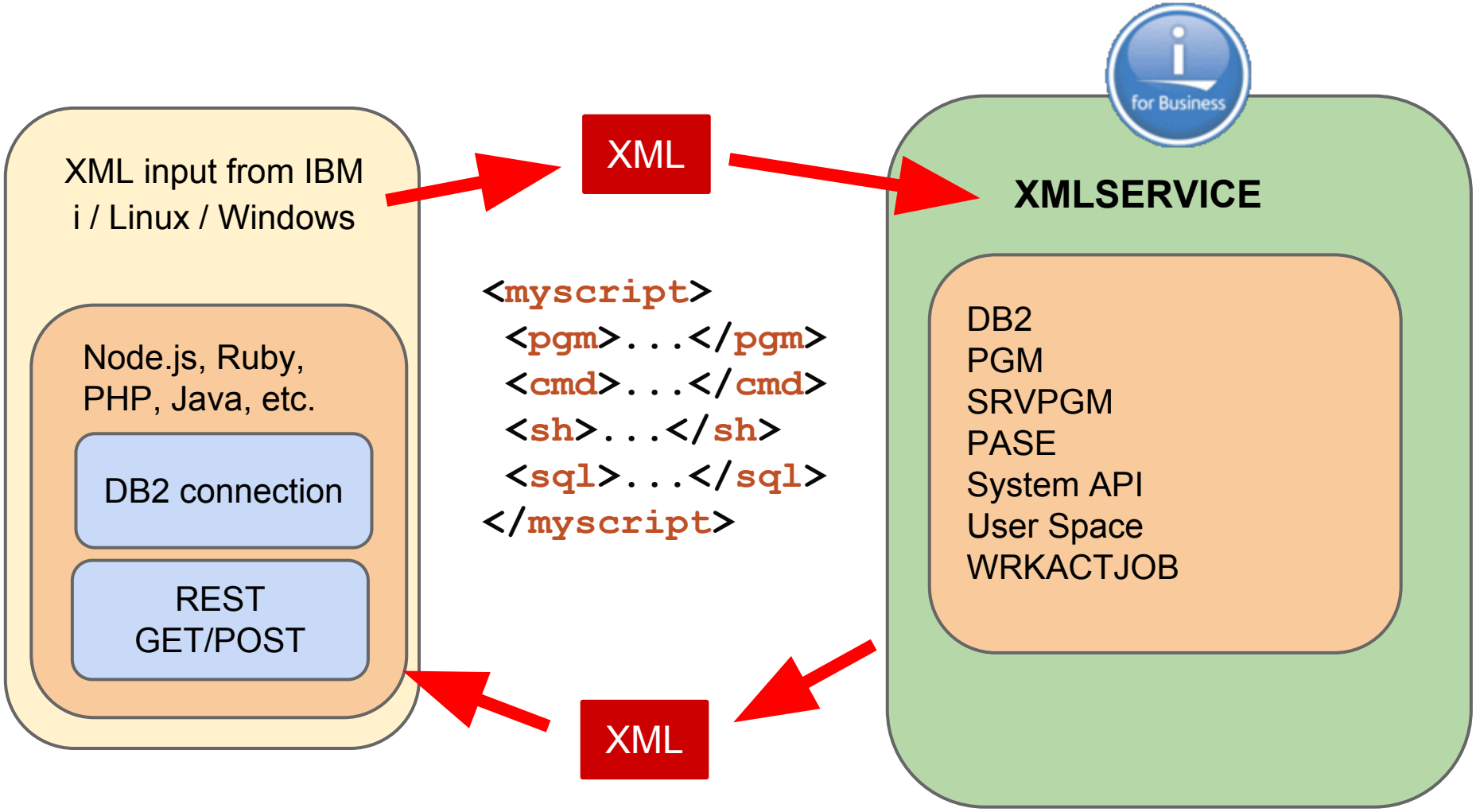
- Delivered with OS as of IBM i 7.1: TR5
- Similar to IBM ToolBox for Java, but better in many ways
- Great for re-use of existing RPG code from ANY language in your enterprise
- Can be invoked via DB2 stored procedure or HTTP
- **Author:** Tony Cairns, IBM Rochester

Make use of all these from Node.js

- DB2 for i – SQL and Native
- *PGM call
- *SRVPGM sub procedure call
- Data Area
- Data Queue
- Message Queue
- Commands
- System values
- Spool files

YoungiProfessionals.com/wiki/XMLSERVICE - Home site
YoungiProfessionals.com/wiki/index.php/NodeJs/NodeJs - Node.js on YiPs

Flow and 10k foot view



itoolkit.js in action

- `itoolkit.js` is an `XMLSERVICE` wrapper.
- Location: `/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit.js`
- It's open source. Take a look at how they put it together!

```
1. var xt = require("/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit");
2. var conn = new xt.iConn("*LOCAL");
3. conn.add(xt.iCmd('RTVJOBA USER(?)'));
4. conn.run(function(str) {
5.   console.log(str);
6. });
```

Output --->

```
1. <?xml version='1.0'?>
2. <myscript>
3.   <cmd exec='rexx' error='fast'>
4.     <success>+++ success RTVJOBA USER(?)</success>
5.     <row>
6.       <data desc='USER'>QUSER</data>
7.     </row>
8.   </cmd>
9. </myscript>
```

node REPL

Type `node` at the command line to start a REPL session. Copy/paste code for quick testing! REPL = **R**ead **E**val **P**rint **L**oop

nodejs.org/api/repl.html - REPL docs

xmlToJson

Javascript deals in JSON, not so much XML. . . enter `xmlToJson` API.

```
conn.run(function(str) {
    result = xt.xmlToJson(str);
    console.log(result[0].data[0].value);
});
```

Before

```
<?xml version='1.0'?>
<myscript>
  <cmd exec='rexx' error='fast'>
    <success>+++ success
             RTVJOBA USER(?)</success>
    <row>
      <data desc='USER'>QUSER</data>
    </row>
  </cmd>
</myscript>
```

After

```
[{ "type":"cmd",
   "success":true,
   "cmd":"RTVJOBA USER(?)",
   "data":[{
     "name":"USER",
     "value":"QUSER"
   }]
}]
```

debugging

```
1.  var conn = new xt.iConn("*LOCAL");
2.  conn.debug(true);
```

Additional output:

```
1.  ============
2.  INPUT XML
3.  ============
4.  <?xml version='1.0'?>
5.  <myscript>
6.  <cmd exec='rexx' error='fast'>RTVJOBA USER(?)</cmd>
7.  </myscript>
```

Still can't determine why it's not working? It's open source! Look at the code. . .

`/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit.js`

Call a *PGM

MYLIB/PGM1

```
1.   dcl-pr pgm1 extpgm;
2.     char1 char(1);
3.     dec1 packed(7:4);
4.   end-pr;
5.   dcl-pi pgm1;
6.     char1 char(1);
7.     dec1 packed(7:4);
8.   end-pi;
9.
10.  char1 = 'C';
11.  dec1 = 321.1234;
12.  return;
```

NOTE: This is 100% free-form RPG, available in IBM i v7.1

The `itoolkit.js` and `XMLSERVICE` support much more complex program calls and even go beyond the PCML/Java Toolbox limitations.

Invoke MYLIB/PGM1

```
1.  var xt = require('/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit');
2.  var conn = new xt.iConn("*LOCAL");
3.
4.  var pgm = new xt.iPgm("PGM1", {"lib":"MYLIB"});
5.  pgm.addParam("","1A");
6.  pgm.addParam("0", "7p4");
7.
8.  conn.add(pgm.toXML());
9.
10. conn.run(function (rsp) {
11.   var results = xt.xmlToJson(rsp);
12.   results.forEach(function(result, index){
13.     result.data.forEach(function(data, index2){
14.       console.log("type:" + data.type + " value:" + data.value);
15.     });
16.   });
17. });
```

Encapsulation

Encapsulate RPG for easy invocation

```
1.  var rpg = require('rpg');
2.  var result = rpg.pgm1("A","0");
3.  result.char1;
4.  result.dec1;
```

rpg.js

```
1.  var xt = require('/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit');
2.  var conn = new xt.iConn("*LOCAL");
3.
4.  function pgm1(char1, dec1){
5.    var results;
6.    var pgm = new xt.iPgm("PGM1", {"lib":"MYLIB"});
7.    pgm.addParam("","1A");
8.    pgm.addParam("0", "7p4");
9.    conn.add(pgm.toXML());
10.   conn.run(function (rsp) {
11.     results = xt.xmlToJson(rsp);
12.   });
13.   return {char1: results[0].data[0].value,
14.           dec1: results[0].data[1].value};
15. }
16. exports.pgm1 = pgm1;
```

sitepoint.com/understanding-module-exports-exports-node-js - Node.js export tutorial

Data Types *(similar to RPG)*

```
1.  type='na' [varying='on|off|2|4'] - character (32A)
2.  <data type='32a'><![CDATA[<i am ranger>]]></data>
3.
4.  type='npn' - packed decimal (12p2)
5.  <data type='12p2'>30.29</data>
6.
7.  type='nsn' - zoned decimal (12s2)
8.  <data type='12s2'>30.29</data>
9.
10. type='nin' - signed integer (5i0, 10i0, 20i0)
11. <data type='10i0'>-30</data>
12.
13. type='nun' - unsigned integer (5u0, 10u0, 20u0)
14. <data type='10u0'>30</data>
15.
16. type='nfn' - floating point (4f2, 8f4)
17. <data type='4f2'>30.34</data>
18. <data type='8f4'>30.34</data>
19.
20. type='nb' - binary HEX char (2b, 400b)
21. <data type='5b'>F0F1F2CDEF</data>
22. <data type='2b'>1FBC</data>
23. <data type='2b'>0F0F</data>
```

youngiprofessionals.com/wiki/index.php/XMLService/XMLSERVICEQuick - Docs

HTTP Connection

- `itoolkit.js` supports DB2 or HTTP connection.
- If 1-tier you don't need authentication
- `iConn(string dataSource[, string user, string password[,object options]])`

Consideration: HTTP slower than DB2

```
1.  var xt = require('/QOpenSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit');
2.  var option = {
3.      host : '192.168.0.10',
4.      port : 8000,
5.      path : '/cgi-bin/xmlcgi.pgm'
6.  };
7.  var conn2 = new xt.iConn('*LOCAL', 'USR1', 'bleh!', option);
```

Reality: Node.js can reside on Amazon/other and call into your IBM i with ease. But why not keep it on IBM i?

Stateful and Private

- Used when IBM i resource will be called many times by the same user profile
- Lasting persistent data needed by the IBM i resource (RPG vars, open files, etc.)

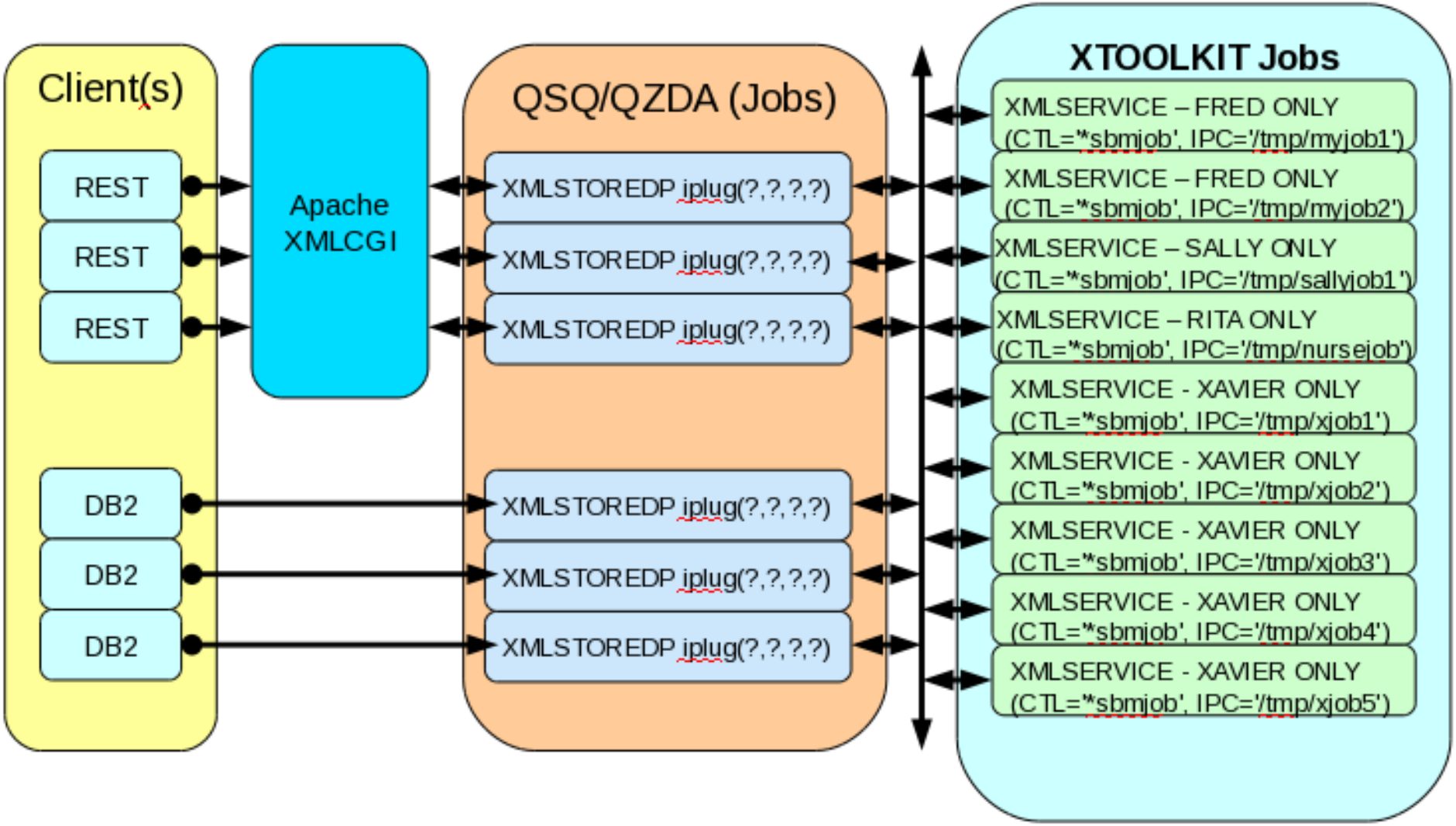


image courtesy of Tony

DB2 Access for i

- **What:** API set for DB2 database manipulation on IBM i
- **Location:** `/QOpenSys/QIBM/ProdData/Node/os400/db2i/`
- **Modules:** `bin/db2i.node` and `lib/db2.js`
- **No ORM yet.** Best bet: github.com/balderdashy/waterline

Example

```
1.  var db = require('/QOpenSys/QIBM/ProdData/Node/os400/db2i/lib/db2');
2.  db.init();
3.  db.conn("*LOCAL");
4.  db.exec("SELECT LSTNAM,CITY FROM QIWS.QCUSTCDT", function(result_set) {
5.    console.log(result_set);
6.  });
7.  db.close();
```

Results

```
1.  [ { LSTNAM: 'Henning ', CITY: 'Dallas' },
2.    { LSTNAM: 'Jones   ', CITY: 'Clay  ' },
3.    { LSTNAM: 'Vine    ', CITY: 'Broton' },
4.    { LSTNAM: 'Johnson ', CITY: 'Helen ' },
5.    { LSTNAM: 'Tyron   ', CITY: 'Hector' },
6.    { LSTNAM: 'Stevens ', CITY: 'Denver' }
7.     . . .
8.  ]
```

DB2 Connection

- **Node.js pgm needs to be on IBM i, or use DB2Connect.**
- **ServerMode** supported - routes SQL statements to separate jobs
- `.init` and `.conn` support callbacks for config (i.e. `serverMode` and `autoCommit`)
- `*LOCAL` will connect to current IBM i. Use `WRKRDBDIRE` to obtain the name of another database.

```
1.  try{
2.    db.init(function(){
3.      db.serverMode(true);
4.    });
5.    db.conn('*LOCAL', function(){
6.      db.autoCommit(true);
7.    });
8.    db.close();
9.  } catch(e) {
10.   console.log(e);
11. }
```

bit.ly/rzadpsqladp_servermode - DB2 ServerMode

DB2 Result Set

- Resulting data is sent to callback function as JSON for easy consumption.
- Metadata is available on connection object via variety of functions: `numRows`, `numFields`, `fieldName`, `fieldWidth`, `fieldType`, `fieldPreceise`, `fieldScale`, `fieldNullable`.

Metadata

```
1.  db.exec('SELECT * FROM MYLIB.MYTABLE', function(result) {
2.    var fieldNum = db.numFields();
3.    console.log('Name | Length | Type | Precise | Scale | Null');
4.    for(var i = 0; i < fieldNum; i++)
5.      console.log('%s | %d | %d | %d | %d | %d',
6.        db.fieldName(i), db.fieldWidth(i), db.fieldType(i),
7.        db.fieldPrecise(i),db.fieldScale(i), db.fieldNullable(i));
8.  });
```

Result processing

```
1.  db.exec('SELECT * FROM MYLIB.MYTABLE', function(result) {
2.    for(var x = 0; x < result.length; x++){
3.      console.log(result[x].COLNAME);
4.    }
5.  });
```

bit.ly/nodejs_db2foriaccess - IBM docs

Future...

waterline 

- Need ORM solution, **Waterline** is our best bet
- DB2 for i not currently supported. **Who's up for getting their hands dirty?**

```
1.  User.create({ name: 'Walter Jr', age: 30 })
2.  .exec(function(err, user) {});
3.
4.  User.findOne({ id: 1 }, function(err, user) {
5.    // Do stuff here
6.  });
7.
8.  User.find()
9.  .where({ id: { '>': 100 }})
10. .where({ age: 21 })
11. .limit(100)
12. .sort('name')
13. .exec(function(err, users) {
14.   // Do stuff here
15. });
16.
17. User.destroy({ id: 1 });
```

github.com/balderdashy/waterline - Waterline repo/docs

On the web

Home site:

youngiprofessionals.com/wiki/index.php/XMLService

youngiprofessionals.com/wiki/index.php/NodeJs/NodeJs

bitbucket.org/litmis/nodejs

Article by Jon Paris and Susan Gantner:

ibmsystemsmag.com/ibmi/developer/rpg/xmlservice_new_life/

ibmsystemsmag.blogs.com/idevelop/xmlservice/

Blog by Tim Rowe (IBM):

iprodeveloper.com/blog/getting-your-data-xml-service

Article by Brian May:

iprodeveloper.com/development/unleash-your-ibm-i-xmlservice

We Have Reached The End!

Aaron Bartell

abartell@krengeltech.com - @aaronbartell

KRENGELTECH

LITMIS.COM - Open Source for i