

Node, native code and Couchnode



Twitter: @ischi

Github: @sideshowcoder

Developer Advocate, Couchbase

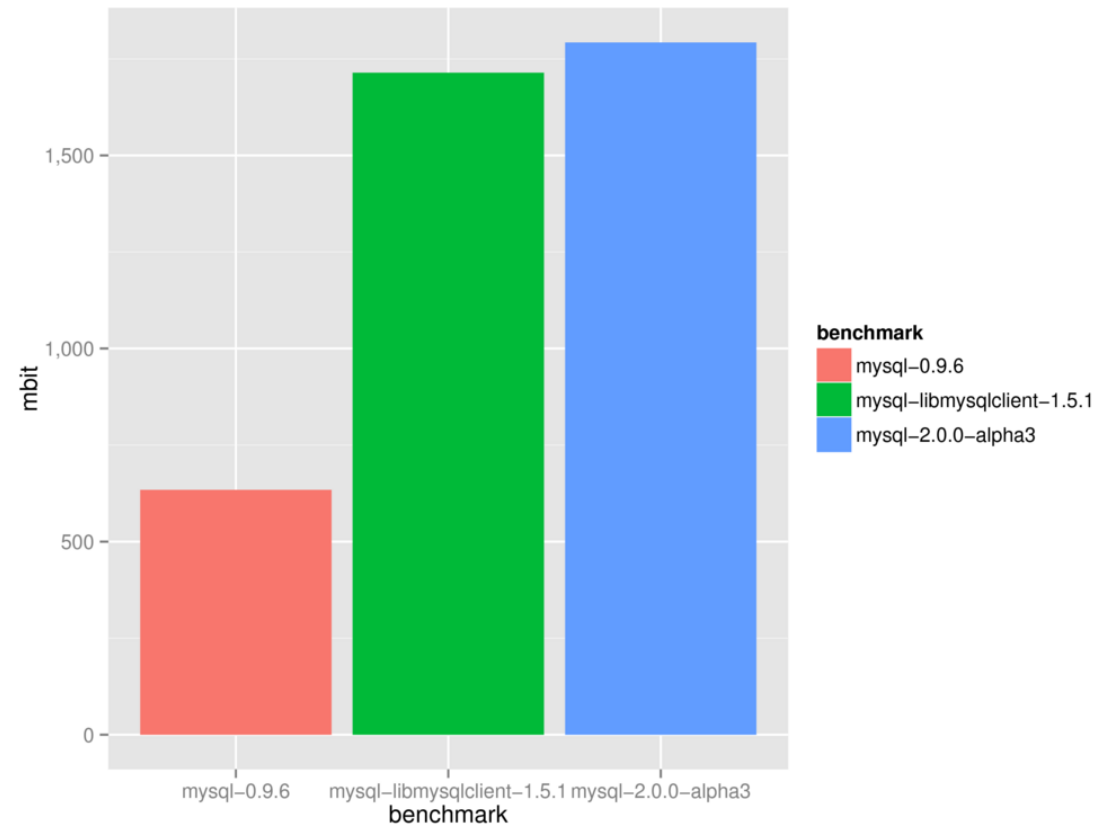
PHILIPP FEHRE

Using C/C++ from node

NODE.JS AND NATIVE CODE

Why would I want to do this?

Of course Speed!



Nope...

Listen to @felixge faster-than-c for more

C is cool!

Java

C

PHP

Ruby

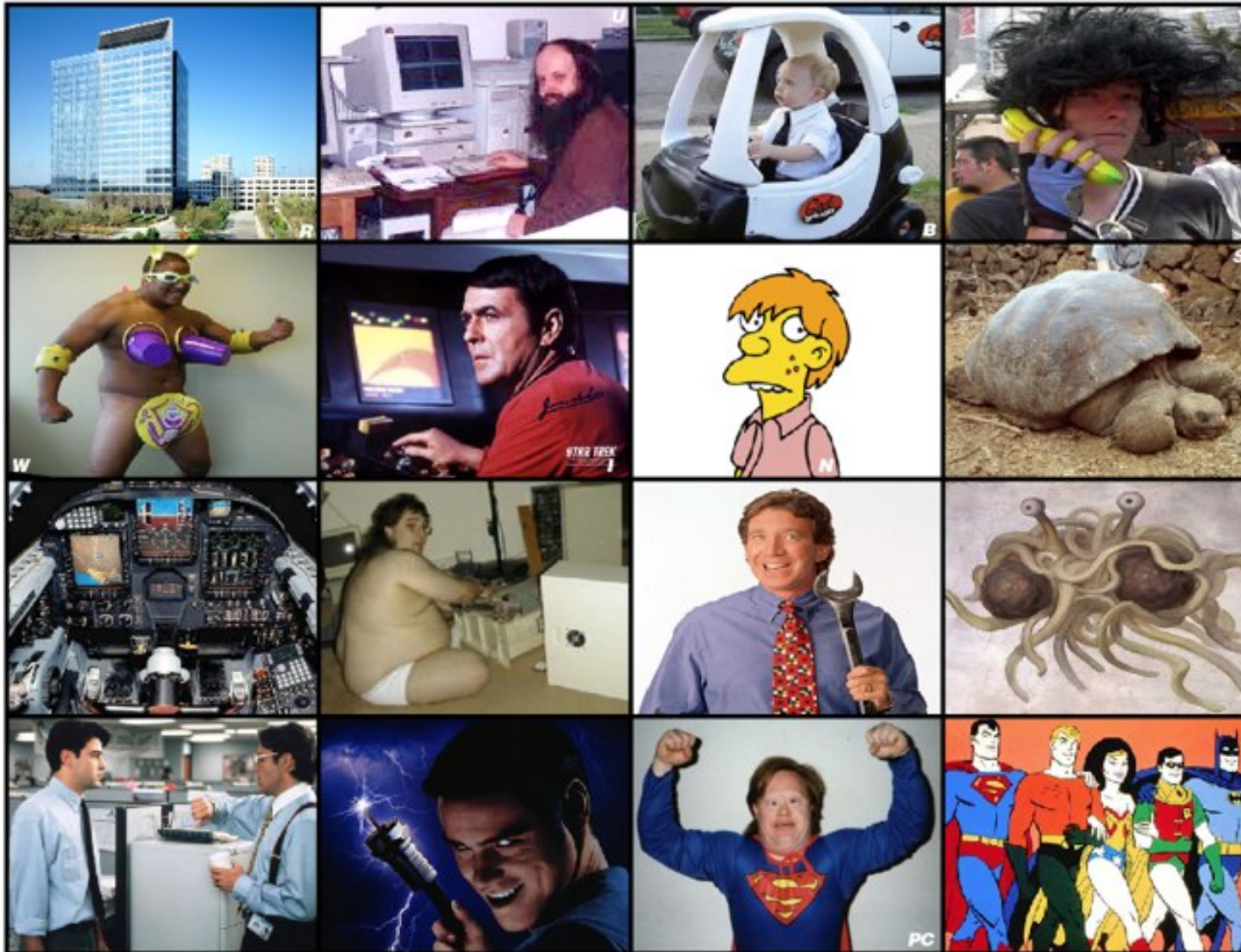
as seen by...

Java fanboys

C fanboys

PHP fanboys

Ruby fanboys



Probably the best reason

USING A C LIBRARY FROM NODE.JS

Couchnode is one of those cases

Based on libcouchbase + wrapper

Quick: What's Couchbase?

A Document Database.



Focus on Performance





Allow flexibility by running JS

[Cluster Overview](#)[Server Nodes](#)[Data Buckets](#)[Views](#)[XDCR](#)[Log](#)[Settings](#)

Data Buckets

Couchbase Buckets

[Create New Data Bucket](#)

Bucket Name	Nodes	Item Count	Ops/sec	Disk Fetches/sec	RAM/Quota Usage	Data/Disk Usage	
▶ default	 1	4	0	0	31.1MB / 128MB	28.5MB / 28.7MB	Documents Views
▶ ncqa_development	 1	6	0	0	31.1MB / 128MB	20.4MB / 20.5MB	Documents Views
▶ ncqa_test	 1	0	0	0	31.1MB / 128MB	20MB / 20MB	Documents Views
▶ todos	 1	20	0	0	31.1MB / 128MB	24.4MB / 24.7MB	Documents Views

Using it from node.js


```
var couchbase = require("couchbase")
var cluster = new couchbase.Cluster()
var bucket = cluster.openBucket("default")

var doc = { store: "json we can", multiple: "values it can have" }

bucket.upsert("my-key", doc, function (err, res) {
  if (err) throw err
  bucket.get("my-key", function (err, res) {
    if (err) throw err
    console.log(res)
    bucket.disconnect()
  })
})
```

node-couch-example\$

[INS]



What happens here?

Or how we finally get to see some C

```
function Bucket(options) {  
  // We normalize both for consistency as well as to  
  // create a duplicate object to use  
  options.dsnObj = connStr.normalize(options.dsnObj);  
  
  var bucketDsn = connStr.stringify(options.dsnObj);  
  var bucketUser = options.username;  
  var bucketPass = options.password;  
  
  this._name = options.dsnObj.bucket;  
  this._username = options.username;  
  this._password = options.password;  
  
  this._cb = new CBpp(bucketDsn, bucketUser, bucketPass);  
}
```

```
var binding = require('./binding');
var connStr = require('./connstr');
var ViewQuery = require('./viewquery');
var N1qlQuery = require('./n1qlquery');
var BucketManager = require('./bucketmgr');

var CONST = binding.Constants;
var CBpp = binding.CouchbaseImpl;
```

```
void CouchbaseImpl::Init(Handle<Object> target)
{
    NanScope();

    Local<FunctionTemplate> t = NanNew<FunctionTemplate>(fnNew);
    t->InstanceTemplate()->SetInternalFieldCount(1);
    t->SetClassName(NanNew<String>("CouchbaseImpl"));
}
```

```

NanScope();

if (args.Length() != 3) {
    return NanThrowError(Error::create("expected 3 parameters"));
}

lcb_error_t err;

lcb_io_opt_st *iops;
lcbuv_options_t iopsOptions;

iopsOptions.version = 0;
iopsOptions.v.v0.loop = uv_default_loop();
iopsOptions.v.v0.startsop_noop = 1;

err = lcb_create_libuv_io_opts(0, &iops, &iopsOptions);
if (err != LCB_SUCCESS) {
    return NanThrowError(Error::create(err));
}

lcb_create_st createOptions;
memset(&createOptions, 0, sizeof(createOptions));
createOptions.version = 3;
if (args[0]->BooleanValue()) {
    createOptions.v.v3.connstr = (char*)_NanRawString(
        args[0], Nan::UTF8, NULL, NULL, 0, v8::String::NO_OPTIONS);
}

```

Seriously I like to show code

But this starts to be a little much

The hello world of extension

Hey some live coding

0.10 vs 0.11

Everything changes, welcome to the
world of Nan

Native Abstractions for Node.js

A header file filled with macro and utility goodness for making add-on development for Node.js easier across versions 0.8, 0.10 and 0.11, and eventually 0.12.

```
{  
  "targets": [  
    {  
      "target_name": "hello",  
      "sources": [ "hello.cc" ],  
      "include_dirs": [ "    },  
  ],  
}
```

```
#include <node.h>
#include <v8.h>
#include <nan.h>

using namespace v8;

NAN_METHOD(world) {
    NanScope();

    NanReturnValue(String::New("hello world"));
}

void init(Handle<Object> target) {
    NODE_SET_METHOD(target, "world", world);
}

NODE_MODULE(hello, init);
```

What could go wrong?

Wrapping C

```
gyp info spawn args [ 'BUILDTYPE=Release', '-C', 'build' ]
  CXX(target) Release/obj.target/murmur3/src/binding.o
  CC(target) Release/obj.target/murmur3/src/murmur3.o
  SOLINK_MODULE(target) Release/murmur3.node
  SOLINK_MODULE(target) Release/murmur3.node: Finished
gyp info ok

nurmur
dyld: lazy symbol binding failed: Symbol not found: __Z19MurmurHash3_x64_128PKvi
jPv
Referenced from: /Users/phil/Source/nurmur/build/Release/murmur3.node
Expected in: dynamic lookup
```

```
#include <node.h>
#include <nan.h>

extern "C" {
    #include "murmur3.h"
}
```

Arguments and callbacks

```
NAN_METHOD(murmur3_x64_128) {  
  NanScope();  
  
  String::Utf8Value dataObject(args[0]->ToString());  
  const char* data = *dataObject;  
  uint32_t seed = args[1]->Uint32Value();  
  Local<Function> cb = Local<Function>::Cast(args[2]);
```



```
char* buffer = (char *)malloc(16);  
memset(buffer, 0, 16);  
MurmurHash3_x64_128(data, seed, dataObject.length(), buffer);  
  
Local<Value> result[2] = {  
    NanNull(),  
    NanNewBufferHandle(buffer, 16)  
};  
  
cb->Call(Context::GetCurrent()->Global(), 2, result);  
  
NanReturnUndefined();
```

Some more couchnode

To finish of...

What happens on a GET

```
NAN_METHOD(CouchbaseImpl::fnGet) {
    CouchbaseImpl *me = ObjectWrap::Unwrap<CouchbaseImpl>(args.This());
    LcbCmd<lcb_get_cmd_st> cmd;
    void *cookie;
    NanScope();

    cmd->version = 0;
    if (!_ParseKey(&cmd->v.v0, args[0])) {
        return NanThrowError(Error::create("bad key passed"));
    }
    if (!_ParseHashkey(&cmd->v.v0, args[1])) {
        return NanThrowError(Error::create("bad hashkey passed"));
    }
    if (!_ParseUintOption(&cmd->v.v0.exptime, args[2])) {
        return NanThrowError(Error::create("bad expiry passed"));
    }
    if (!_ParseUintOption(&cmd->v.v0.lock, args[3])) {
        return NanThrowError(Error::create("bad locked passed"));
    }
}
```

```
if (!_ParseCookie(&cookie, args[4])) {  
    return NanThrowError(Error::create("bad callback passed"));  
}  
  
lcb_error_t err = lcb_get(me->getLcbHandle(), cookie, 1, cmd);  
if (err) {  
    return NanThrowError(Error::create(err));  
}  
  
NanReturnValue(NanTrue());  
}
```

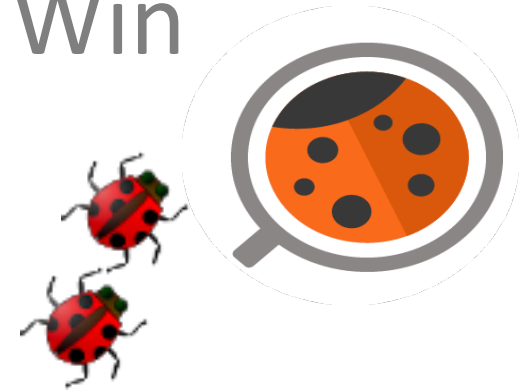
Why is cas an Array?

```
Handle<Value> Cas::CreateCas(uint64_t cas) {  
    Local<Object> ret = NanNew<Object>();  
    uint64_t *p = new uint64_t(cas);  
    ret->SetIndexedPropertiesToExternalArrayData(  
        p, v8::kExternalUnsignedIntArray, 2);  
    NanMakeWeakPersistent(ret, p, casDtor);  
    return ret;  
}
```

Couchbase Server 3.0 Beta Bug Bash

Download, Find, Report, Win

- **DOWNLOAD Couchbase Server 3.0 Beta**
- **Run some workload, play with it, FIND a bug**
- **If you find a bug, REPORT it!**
 - All bugs count – anything from correctness, to performance, to usability, to docs
- **WIN Prizes!!**



<http://www.couchbase.com/bugbash>

Contest ends Aug 31st 2014

Come to spanconf.io

SPAN CONFERENCE

LONDON, 28TH OCTOBER 2014

SINGLE TRACK DEVELOPER CONFERENCE ABOUT SCALING

061

DAYS

17

HRS

33

MIN

54

SEC

⊕ BUY TICKETS

Thank you!

Any questions?

Links

- <https://github.com/kkaefer/node-cpp-modules>
- <https://github.com/TooTallNate/node-gyp>
- <https://github.com/rvagg/nan>
- <https://github.com/couchbase/couchnode>
- <http://www.couchbase.com/communities/nodejs>