



Your company's executives have decided to re-write the company's old web application using new technologies.

While sipping martinis at the country club, they want to be able to say their platform is “api -based” and uses “no sql” -- after all, the most buzz words wins, right?

*Now you've got to make sure all the functionality in the old app works in the new one, **but how?***



What are our Options?

Solution	Challenge
Requirements to Identify Gaps	Requirements may not be complete or accurate
Manual Test Plans	May not be accurate
Existing Functional/UI Tests	Likely don't work... UI usually developed last
Memory, will power, & brow sweat!	Error-prone, grumpiness
Developer Unit Tests	Incomplete - Blinders on



What are our Options?

Solution	Challenge
Requirements to Identify Gaps	Requirements may not be complete or accurate
Manual Test Plans	May not be accurate
Existing Functional/UI Tests	Likely don't work... UI usually developed last
Memory, will power, & brow sweat!	Error-prone, grumpiness
Developer Unit Tests	Incomplete - Blinders on
Integration Tests	Know how



BeaufortFairmont

Integration Testing: Why, When and *How*?

Paul Merrill
Founder Beaufort Fairmont, LLC
@dpaulmerrill
beaufortfairmont.com



BeaufortFairmont

This Presentation

Codebase for this presentation and updates to these slides can be found on github at:

<http://github.com/dpm3354/IntegrationTesting>



Demo Install

- Download & Install Python 2.7
 - <http://www.python.org/getit/>
- Install PIP
 - <http://www.pip-installer.org/en/latest/installing.html#install-or-upgrade-pip>
- Install RobotFramework
 - *pip install robotframework*
- Install RequestLibrary
 - *pip install -U robotframework-requests*
- Install pymysql
 - *pip install pymysql*
 - *pip install robotframework-databaselibrary*



Levels of Testing

- **Unit Tests** – For Developers, Influences Design of Code, Future Assurance, Maintenance, Test First
- **Integration Tests** – Subsystems, Intra/Intersystem, before/during/after development
- **Acceptance** – End User is usually actor, existing system



Integration Testing

- Do systems or subsystems work together?
- May interact with resources – DB, File System
- Intimate with Codebase (white box)
- Simulators, Mocks, etc



BeaufortFairmont

Where?

System Under Test (SUT)

Component A

Component B

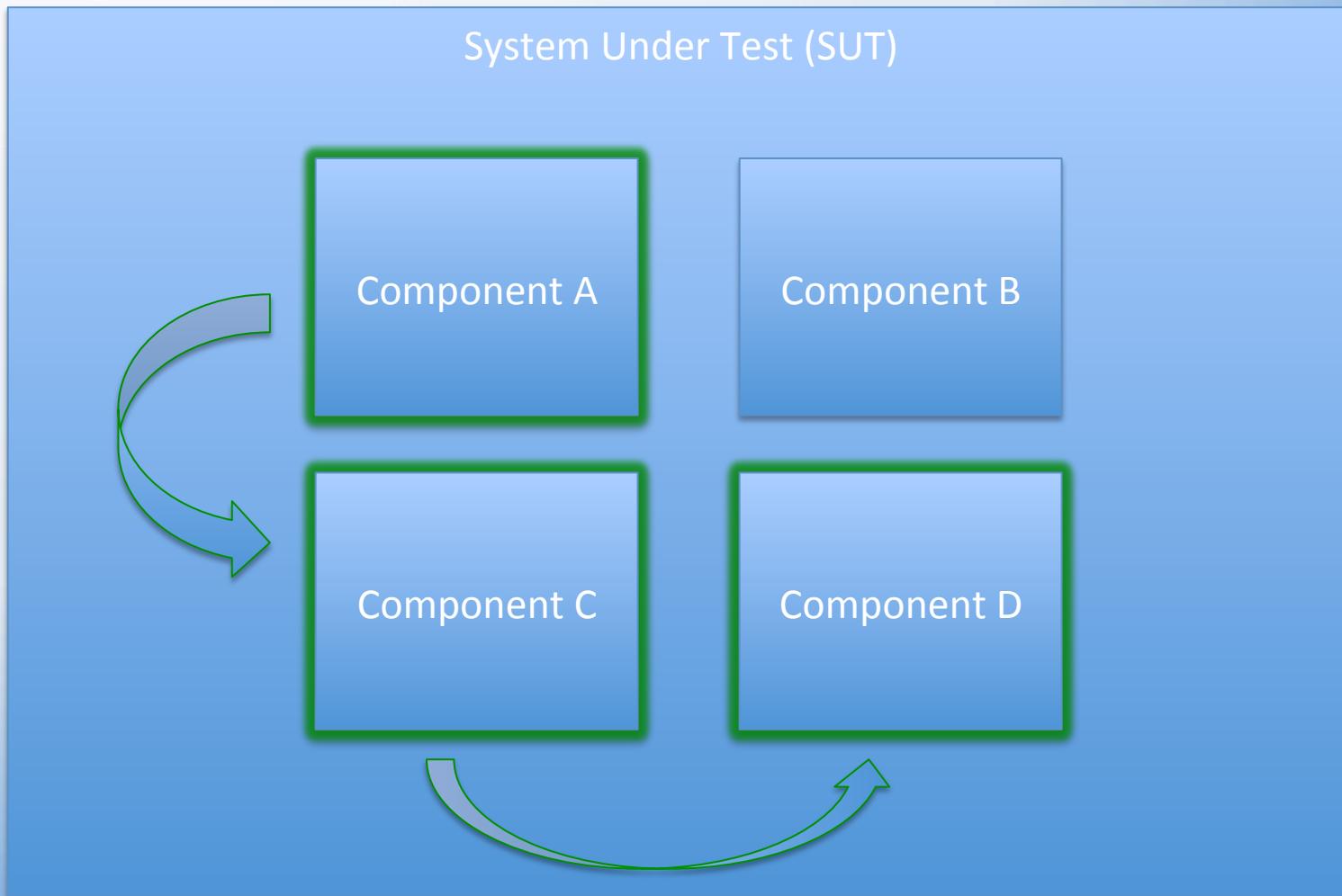
Component C

Component D



BeaufortFairmont

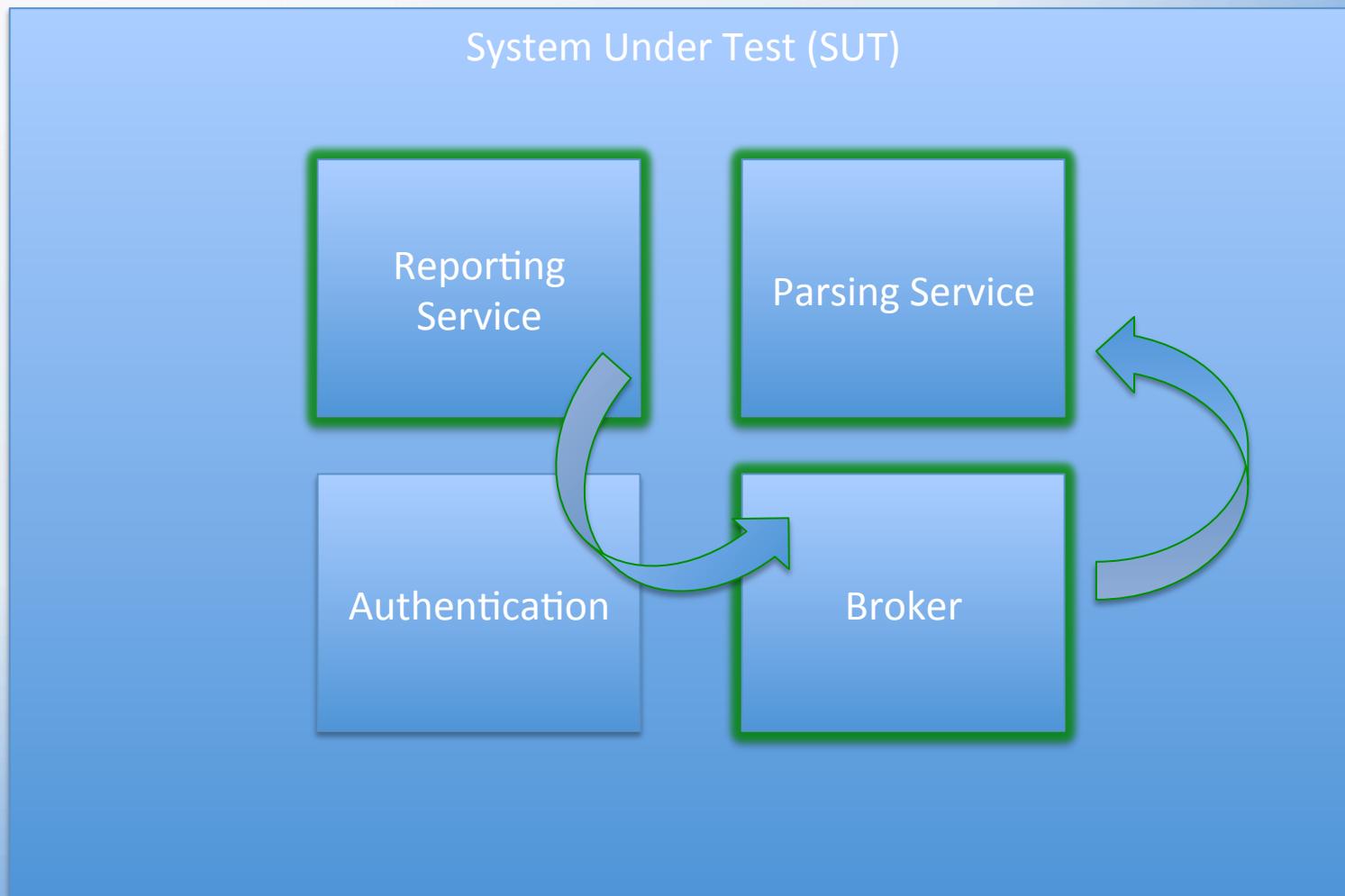
Here





BeaufortFairmont

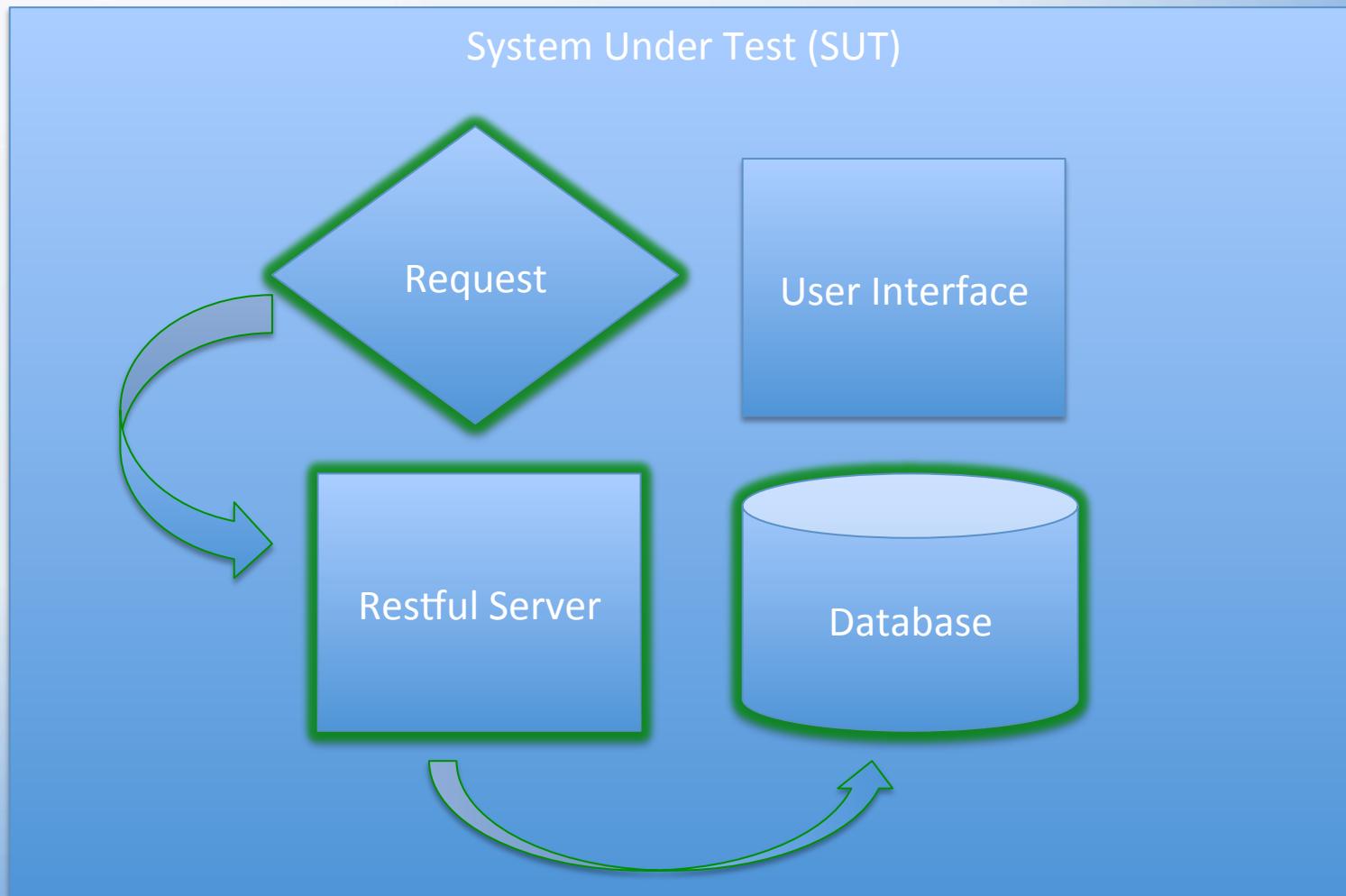
Or Here...





BeaufortFairmont

More Specifically, Here...





BeaufortFairmont

Our Legacy Application

PHP Cellar Application by @ccoenraets

- Restful API
- PHP
- MySQL
- Apache



BeaufortFairmont

Our Testing

Test Cases:

- Make Request
- Check Results
- Check DB
- Cross-Check

Framework:

- RobotFramework



Make Request – API.txt

```
*** Settings ***
```

```
Library Collections
```

```
Library RequestsLibrary
```

```
Library json.JSONEncoder
```

```
*** Variables ***
```

```
${host} 10.0.1.27
```

```
${port} 8888
```

```
${url} http://${host}:${port}
```

```
${path} /api/wines
```

```
*** Test Cases ***
```

```
Check Status
```

```
    Create Session wines ${url}
```

```
    ${resp}=  Get wines ${path}
```

```
    should be equal as strings ${resp.status_code} 200
```



BeaufortFairmont

Run Our Test...

```
$ pybot API.txt
```



API - Sanity Check...

Wines Is Not Empty

```
Create Session wines ${url}
${resp}= Get wines ${path}
${json}=      To JSON ${resp.content}

should not be empty ${json['wine'][0]}
```



API – Inspect Results

```
Wines Contains 12 wines
```

```
Create Session wines ${url}  
${resp}= Get wines ${path}  
${json}= To JSON ${resp.content}
```

```
length should be ${json['wine']} 12
```



API – Check All Wines

```
*** Settings ***
# ...

Test Template Check for Wine


*** Test Cases ***
Check for Block Nine 0 BLOCK NINE
Check for Bodega Lurton 1 BODEGA LURTON
# ...


*** Keywords ***
Check for Wine
    [Arguments]  ${index}  ${winename}
    Create Session  wines  ${url}
    ${resp}=  Get  wines  ${path}
    ${json}=  To JSON  ${resp.content}
    should be equal  ${json['wine'][${index}]['name']}  ${winename}
```



Check the DB

*** Settings ***

library DatabaseLibrary
library pymysql
Suite Setup Connect To Database @{database}
Resource CommonVariables.txt

*** Variables ***

@{database} pymysql cellar root root \${dbhost} \${dbport}

*** Test Cases ***

Check Wine Count

Row Count Is Equal To X select * from wine; 12

Check for Chateau

Check if exists in database select * from wine where name='CHATEAU DE SAINT COSME';



Check All Wines in DB

```
*** Settings ***
# ...
Test Template    Check for Wine

*** Test Cases ***
Check for Wine    CHATEAU DE SAINT COSME
Check for Lan Rioja    LAN RIOJA CRIANZA
# ...

*** Keywords ***
Check for Wine
[Arguments]  ${winename}
    Check if exists in database    select * from wine where name='${winename}';
```



Cross Check

```
#...
*** Test Cases ***
Check for 0th Wine      0
Check for 1st Wine      1
#...
Check for 11th Wine     11

*** Keywords ***
Check for Wine
    [Arguments]  ${index}

    Create Session  wines  ${url}
    ${resp}=  Get  wines  ${path}
    ${json}=  To JSON  ${resp.content}
    LOG  Checking for ${json['wine'][${index}]['name']}  WARN
    Check if exists in database  select * from wine where name='${${json['wine'][${index}]['name']}';


```



BeaufortFairmont

New Application

“nodecellar” by ccoenraets

<https://github.com/ccoenraets/nodecellar>

- Restful API
- Javascript
- NodeJS
- MongoDB



API Tests

```
# CommonVariables.txt

### API Variables ####
${host} 10.0.1.27
${port} 3000

${url} http://${host}:${port}
${path} /wines
```



Notice Failures!

Wines Is Not Empty

| FAIL |

Resolving variable '\${json['wine'][0]}' failed: TypeError: list indices must be integers, not str

Wines Contains 12 wines

| FAIL |

Resolving variable '\${json['wine']}' failed: TypeError: list indices must be integers, not str

API-2

| FAIL |

3 critical tests, 1 passed, 2 failed

3 tests total, 1 passed, 2 failed



BeaufortFairmont

What Now?

- Present to User, Stakeholder
- Bug it
- Move on
 - Our job is to give feedback not proxy the user/owner



Updated Test

Wines Is Not Empty

```
Create Session wines ${url}
${resp}= Get wines ${path}
${json}=      To JSON ${resp.content}

should not be empty ${json[0]}
```



BeaufortFairmont

Notice Failures!

```
Wines Contains 12 wines
| FAIL |
Length of '[{u'picture': u'saint_cosme.jpg', u'description':
...
u'WATERBROOK'}]' should be 12 but is 24
```



BeaufortFairmont

Data Changed

- Migration?
- Report & Move On



Download and Install

- MongoDBLibrary (python version)
 - [https://github.com/
iPlantCollaborativeOpenSource/Robotframework-
MongoDB-Library](https://github.com/iPlantCollaborativeOpenSource/Robotframework-MongoDB-Library)
- pip install pymongo
- sudo python setup.py install
- Added mongodlibrary to /Library/Python/
2.7/site-packages (Mac)



Modify Database-Datadriven

```
*** Settings ***
```

```
Library           MongoDBLibrary
```

```
#...
```

```
Suite Setup      Connect To Mongodb    @{database}
```

```
Test Template    Check for Wine
```

```
#...
```

```
*** Keywords ***
```

```
Check for Wine
```

```
    [Arguments]  ${winename}
```

```
    ${wine}=  Retrieve Mongodb Records With Desired Fields  winedb
wines  {"name": "${winename}"}  name  TRUE
    should contain  ${wine}  ${winename}
```



Finally, we...

- Made Requests
- Checked Results
- Checked DB
- Cross-Checked
- Ran Tests on Both Applications (with minor mods)

Found Changes to:

- Path
- API results
- Data
- All without the UI!



BeaufortFairmont

Integration Testing: Why, When and *How*?

Paul Merrill
Founder Beaufort Fairmont, LLC
@dpaulmerrill
beaufortfairmont.com