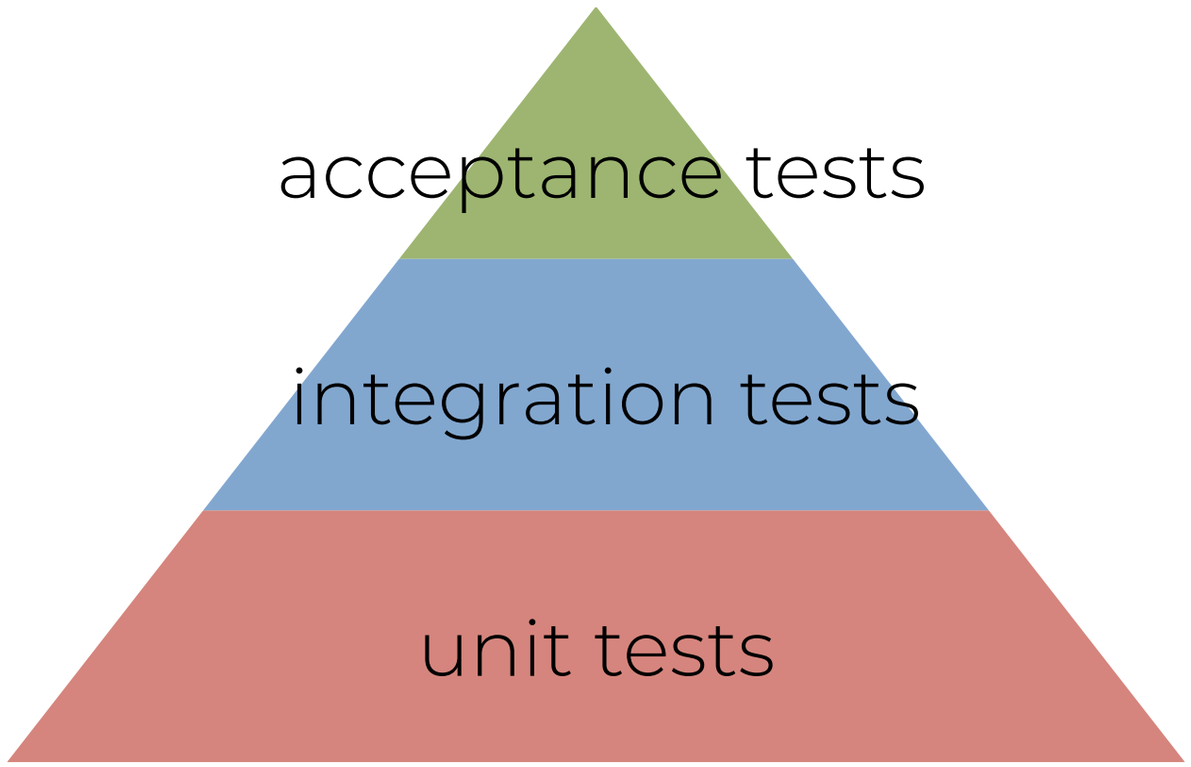
test pyramid



(traditional) test pyramid

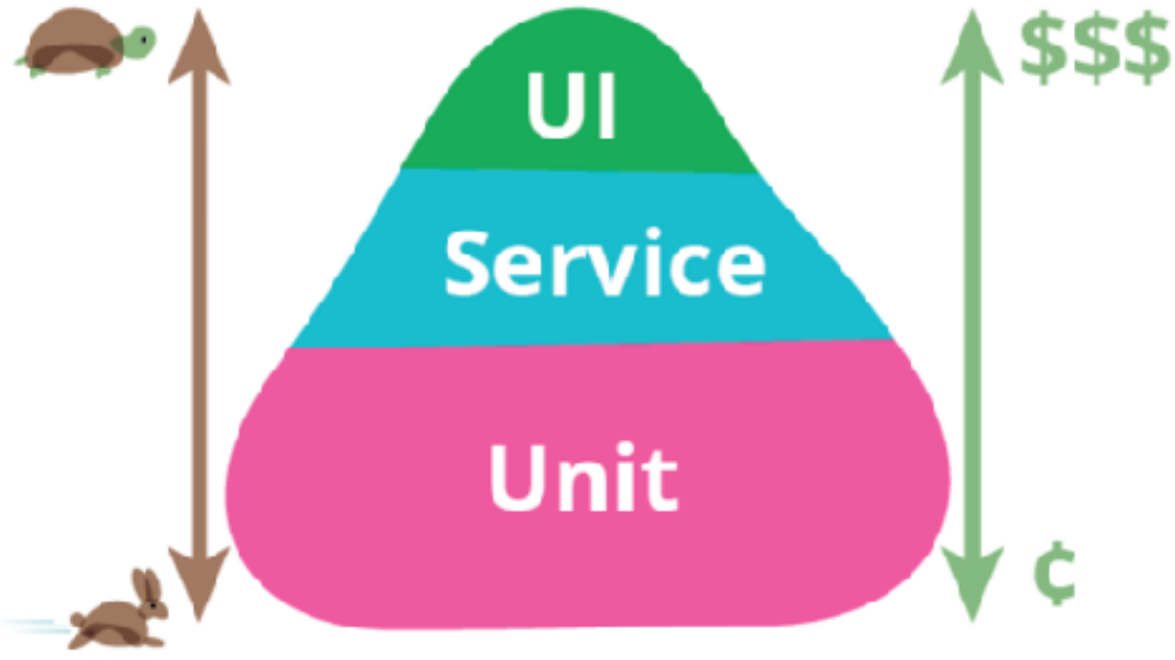
MARTINFOWLER.COM

Intro Videos Design Agile Refactoring FAQ About Me All Sections ThoughtWorks

TestPyramid

Martin Fowler
1 May 2012

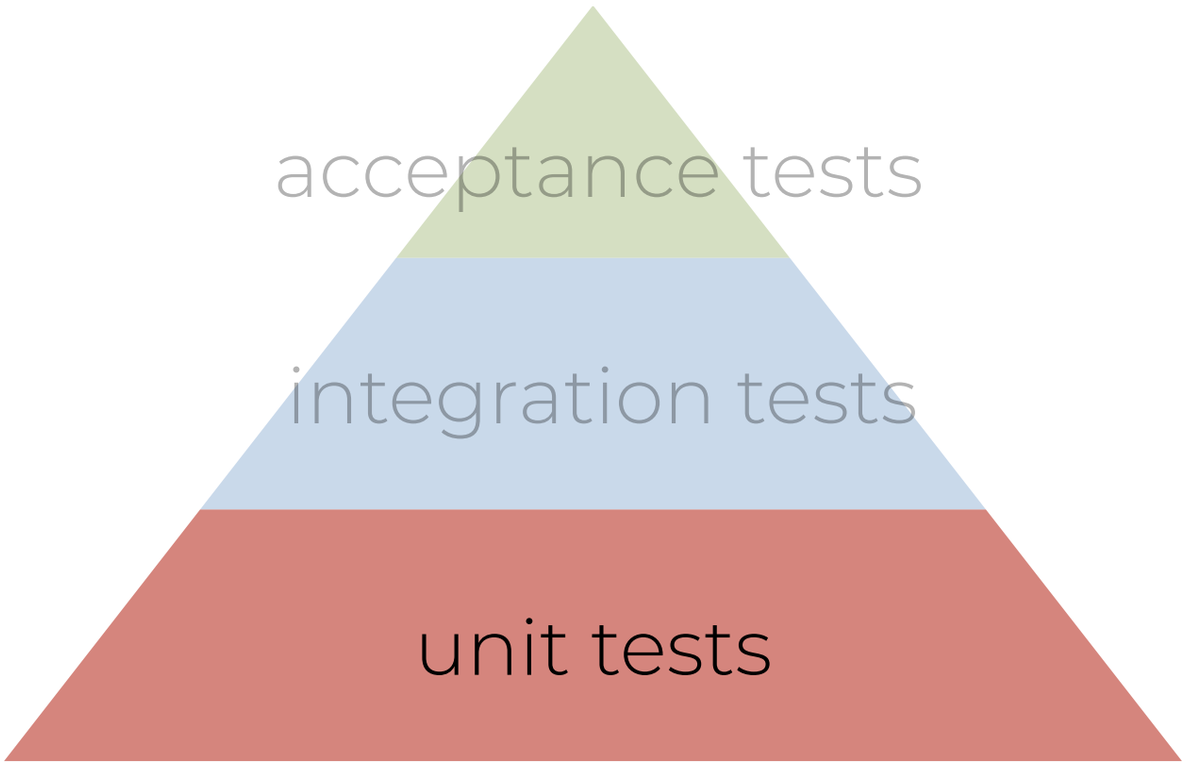
The test pyramid is a way of thinking about different kinds of automated tests should be used to create a balanced portfolio. Its essential point is that you should have many more low-level UnitTests than high level BroadStackTests running through a GUI.



Find similar articles at the tag

testing

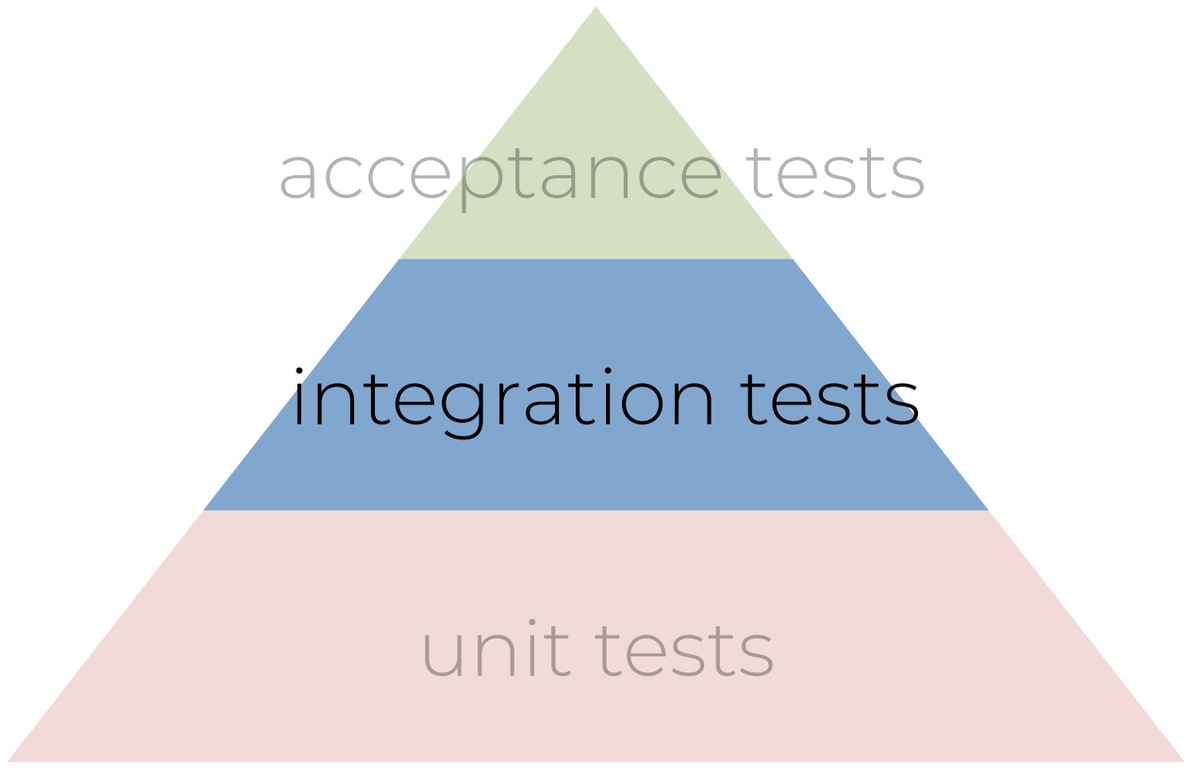
test pyramid



do our objects do the right thing?
are they easy to work with?

```
const APP_ROOT = '../../';
const ironBorn = require(`${APP_ROOT}/lib/ironBorn`);
const expect   = require('chai').expect;

describe(`What's dead`, () => {
  it(`May never die`, () => {
    expect(ironBorn.die({ isDead: true })).to.be.false;
  });
});
```

test pyramid



does our code work against code we can’t change?

```
module.exports.handler = function(event, context, cb) {
  cb(null, { is_here: true });
}
```

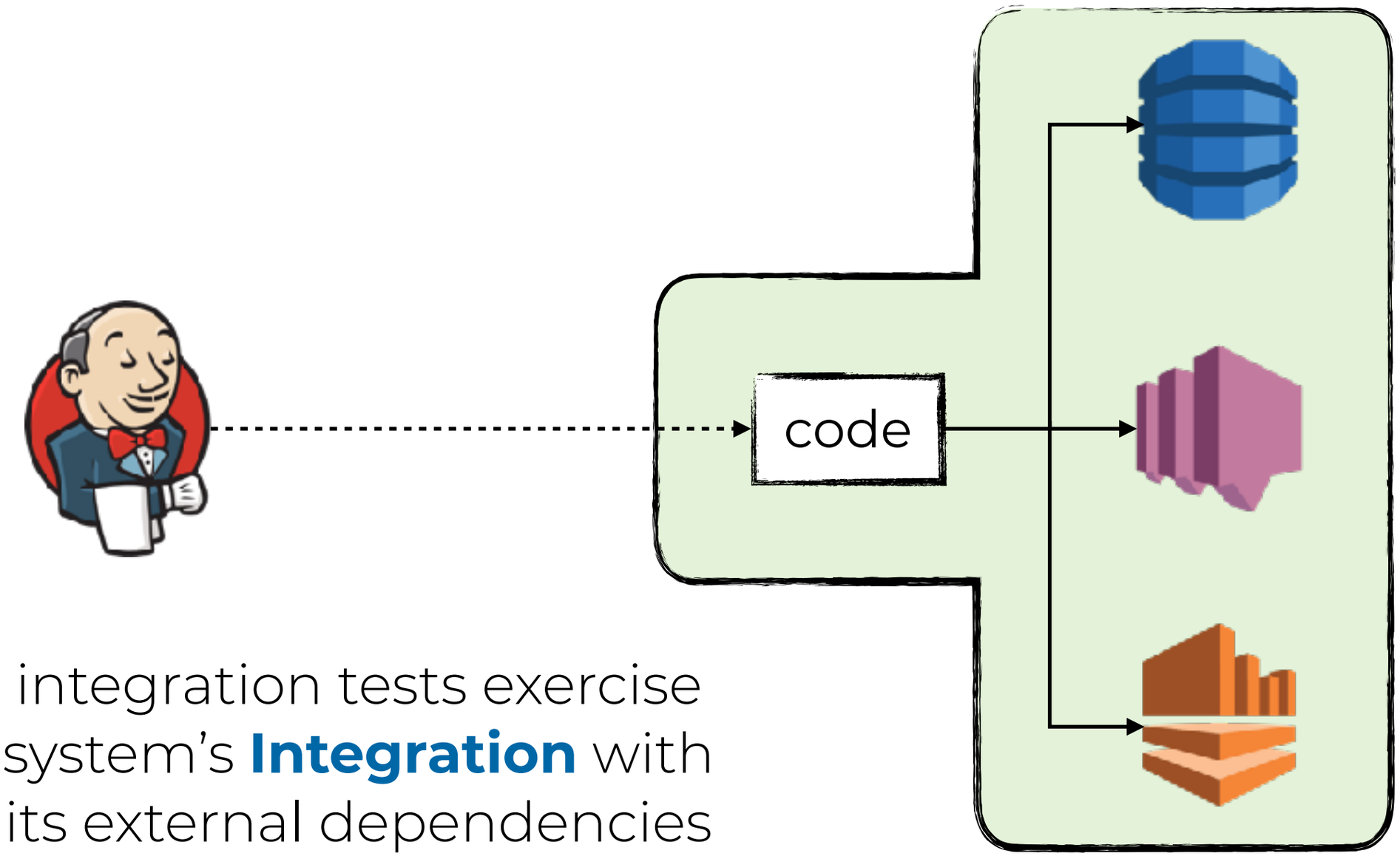
handler

```
module.exports.handler = function(event, context, cb) {
  cb(null, { is_here: true });
}

const handler = require('../../functions/ned').handler;
const expect  = require('chai').expect;

descibe('winter', () => {
  it('should be here', done => {
    let context = {
      // .. request id, etc.
    };

    let event = {
      // eg. mock an API Gateway invocation event
    };

    let cb = function(err, result) {
      expect(err).to.be.null;
      expect(result.is_here).to.be.true;

      done(); // ends test with success
    };

    handler(event, context, cb);
  });
});
```

handler

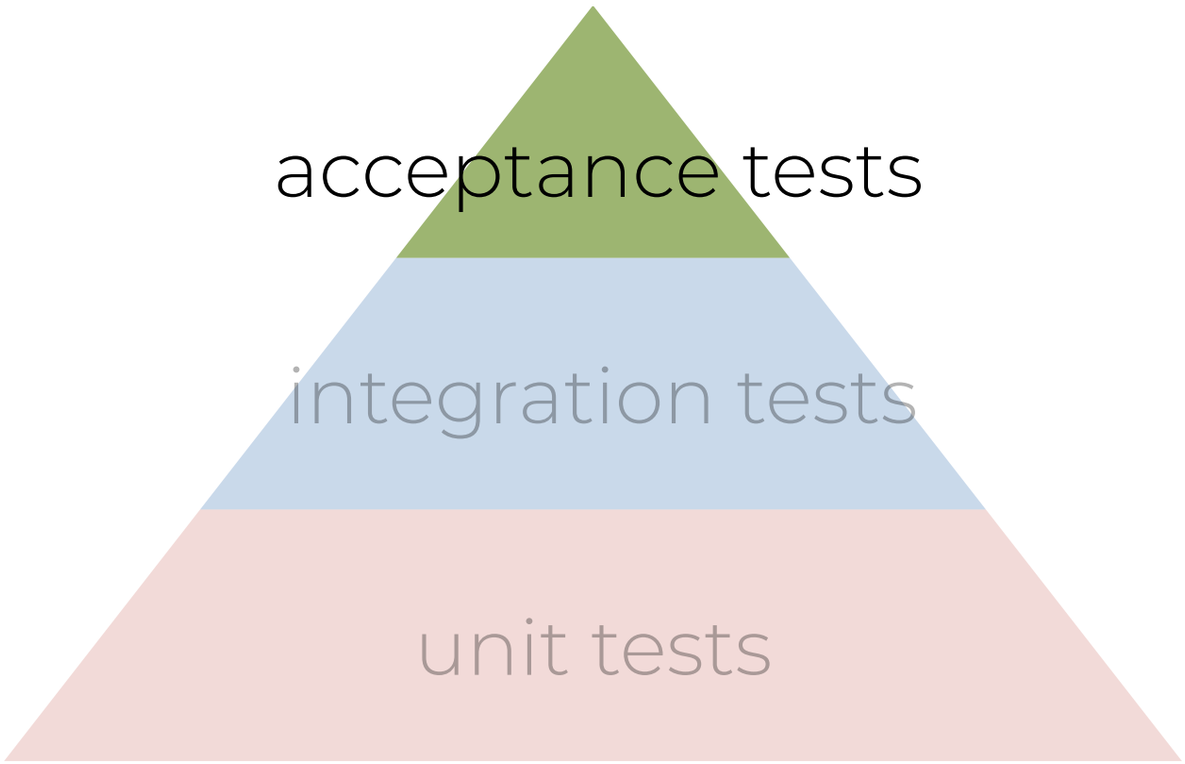
test by invoking the handler

integration tests



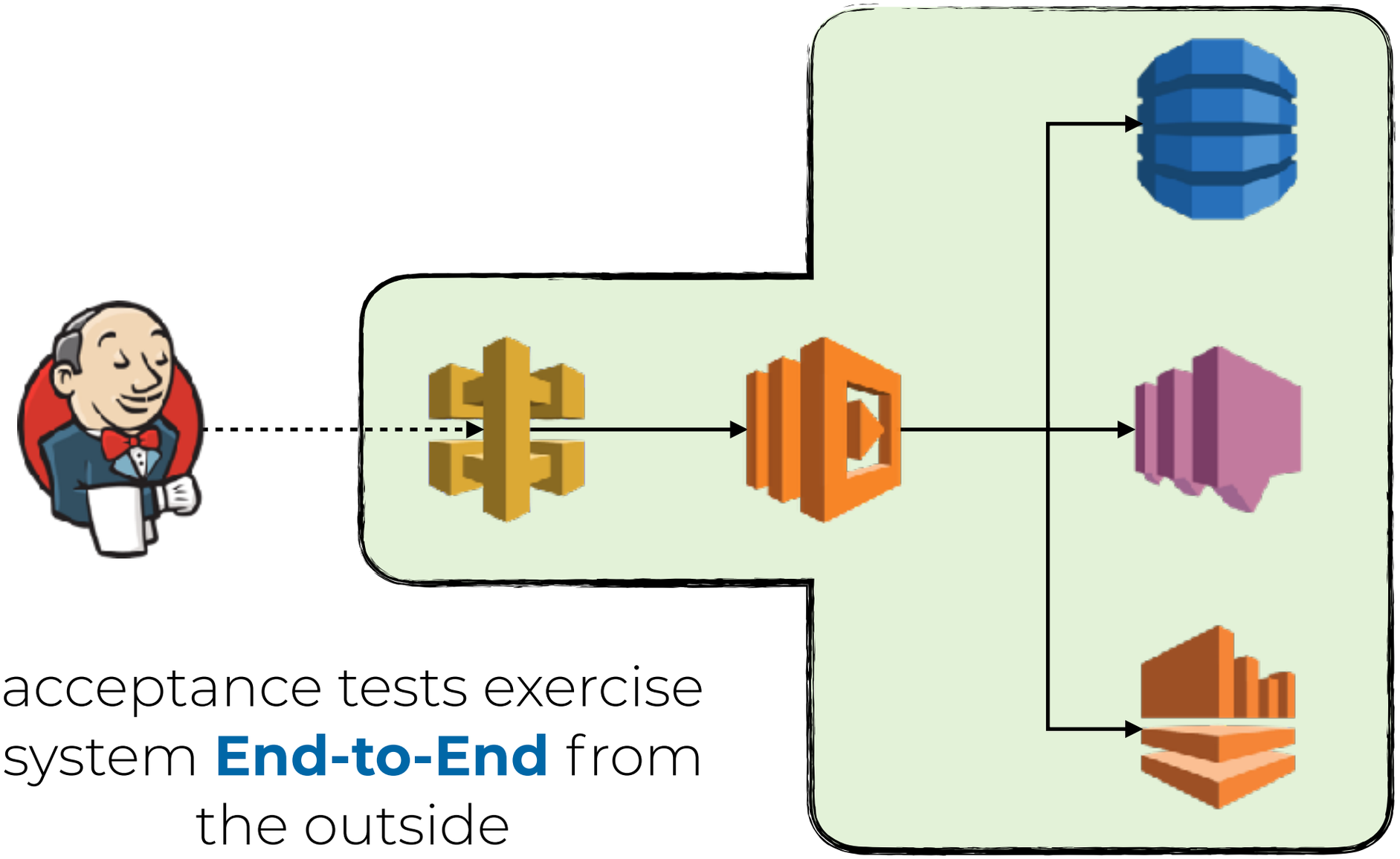
integration tests exercise system's **Integration** with its external dependencies

test pyramid



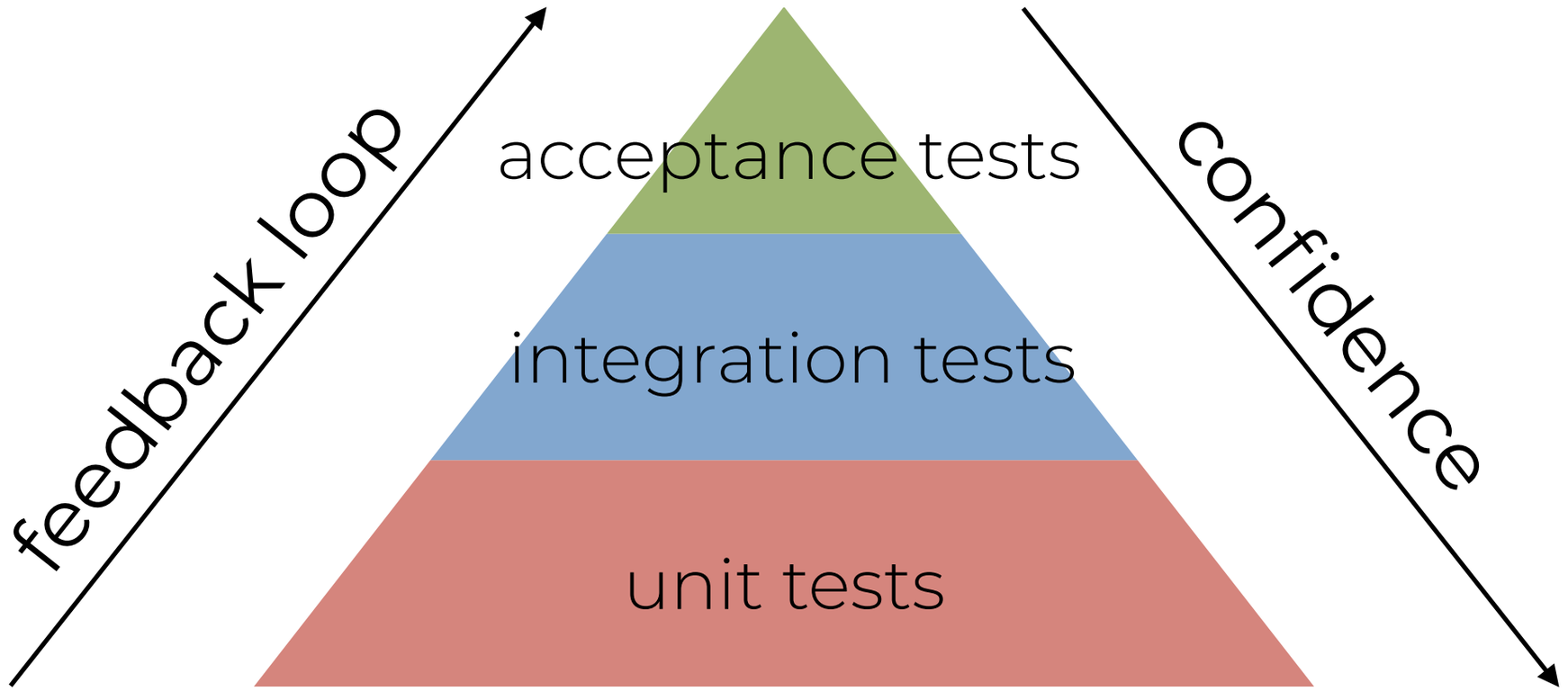
does the whole system work?

acceptance tests

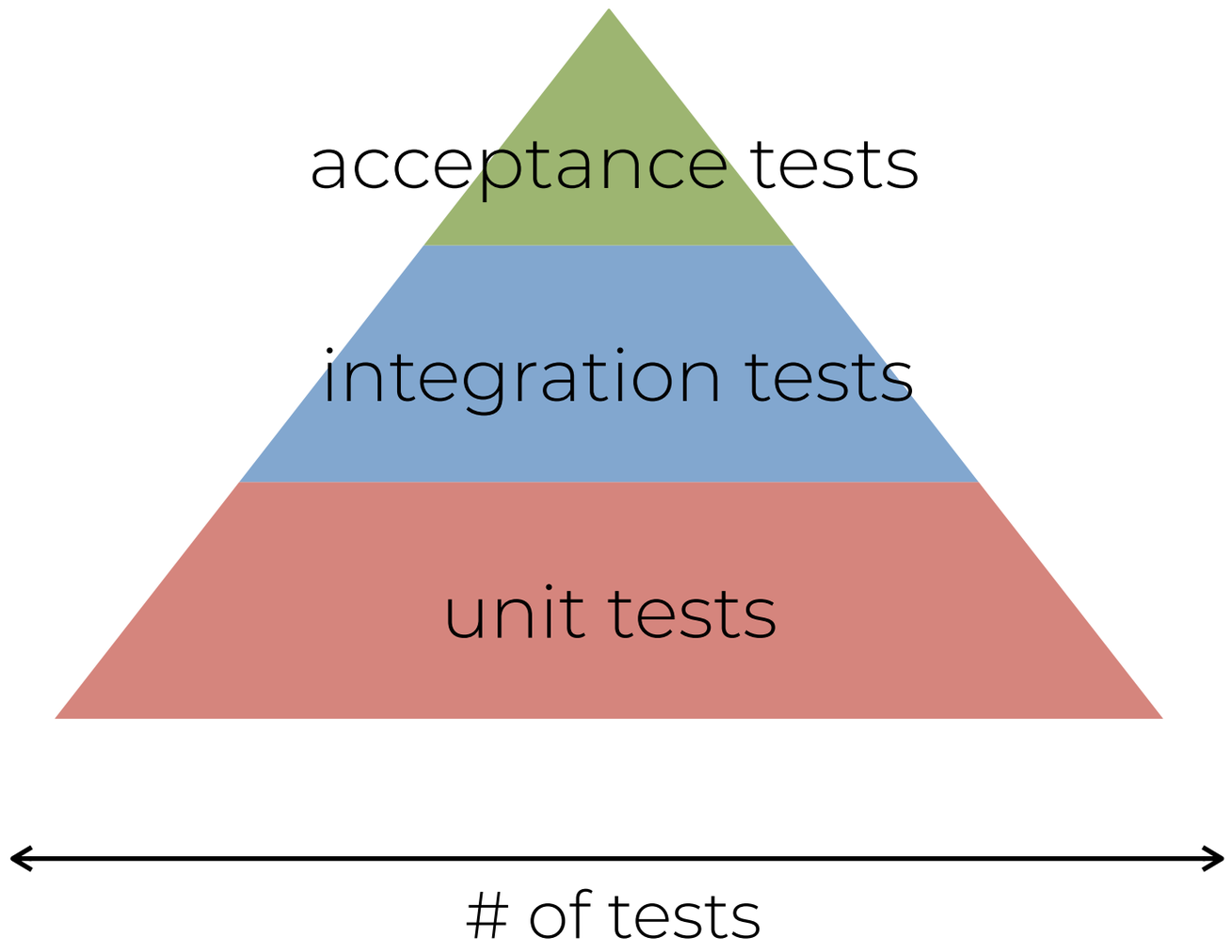


acceptance tests exercise system **End-to-End** from the outside

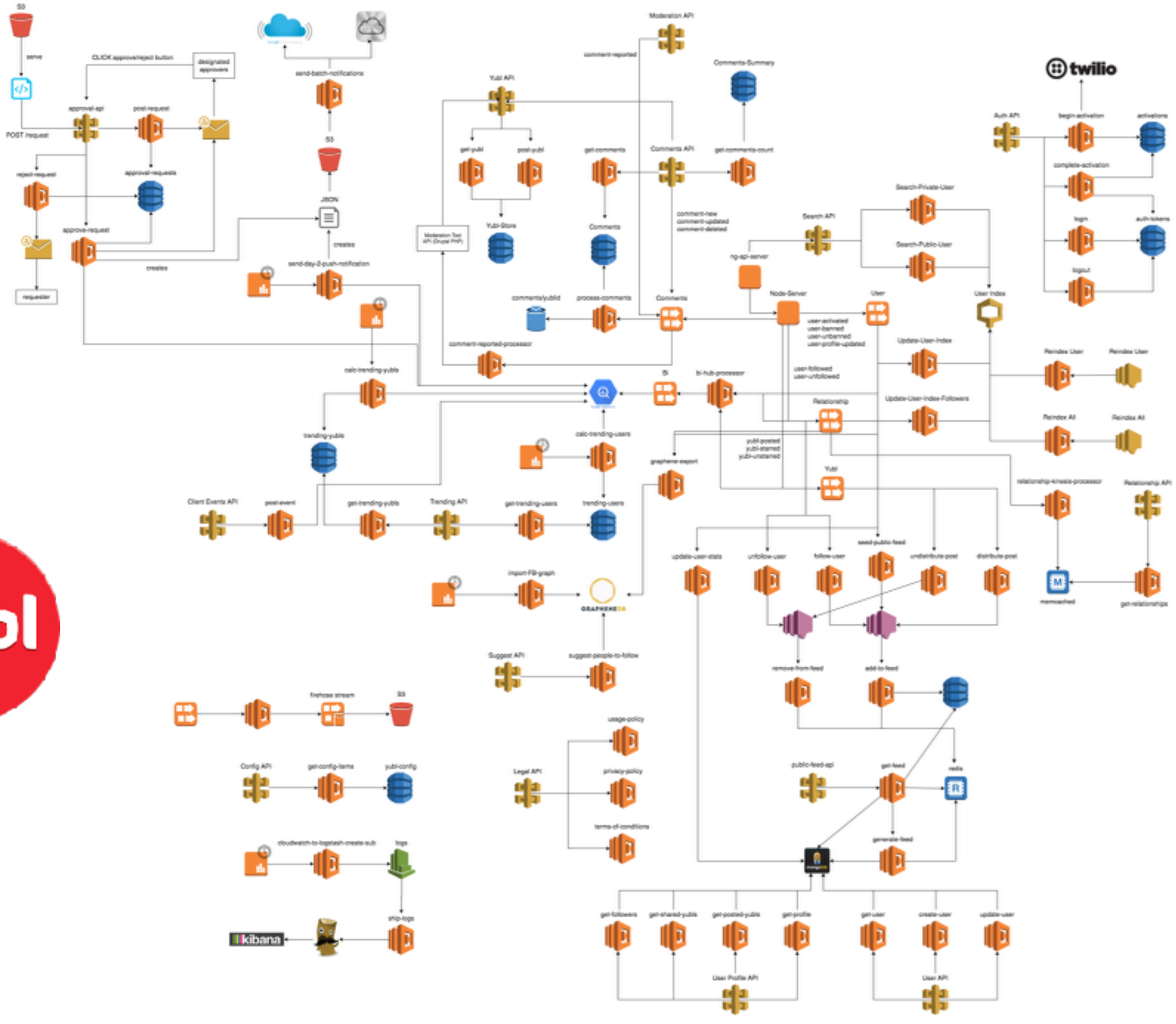
test pyramid



test pyramid



yubl


S3
serve
POST request
CLICK approve/reject button
designated approvers
approval-api
post-request
reject-request
approval-requests
approve-request
requester
creates
send-batch-notifications
S3
JSON
creates
send-day-2-push-notification
Moderation API
comment-reported
Yubl API
get-yubl
post-yubl
Yubl-Store
Moderation Tool API (Drupal PHP)
Comments-Summary
get-comments-count
Comments API
get-comments
Comments
comment-new
comment-updated
comment-deleted
comments/yublId
process-comments
Comments
comment-reported-processor
twilio
Auth API
begin-activation
activations
complete-activation
login
auth-tokens
logout
Search API
Search-Private-User
Search-Public-User
ng-api-server
Node-Server
User
User Index
user-activated
user-banned
user-unbanned
user-profile-updated
user-followed
user-unfollowed
Update-User-Index
Update-User-Index-Followers
Reindex User
Reindex User
Reindex All
Reindex All
Relationship
yubl-posted
yubl-starred
yubl-unstarred
Yubl
relationship-kinesis-processor
Relationship API
memcached
get-relationships
graphene-export
calc-trending-yubls
trending-yubls
calc-trending-users
trending-users
Client Events API
post-event
get-trending-yubls
Trending API
get-trending-users
import-FB-graph
GRAPHENEDB
Suggest API
suggest-people-to-follow
update-user-stats
unfollow-user
follow-user
seed-public-feed
undistribute-post
distribute-post
remove-from-feed
add-to-feed
public-feed-api
get-feed
redis
generate-feed
firehose stream
S3
Config API
get-config-items
yubl-config
Legal API
usage-policy
privacy-policy
terms-of-conditions
cloudwatch-to-logstash-create-sub
logs
ship-logs
kibana
get-followers
get-shared-yubls
get-posted-yubls
get-profile
User Profile API
get-user
create-user
update-user
User API
fb
bi-hub-processor

serverless

observation #1

you use more managed services
when working with AWS Lambda

serverless

observation #2

most Lambda functions are simple, and have a single purpose

serverless

conclusion #1

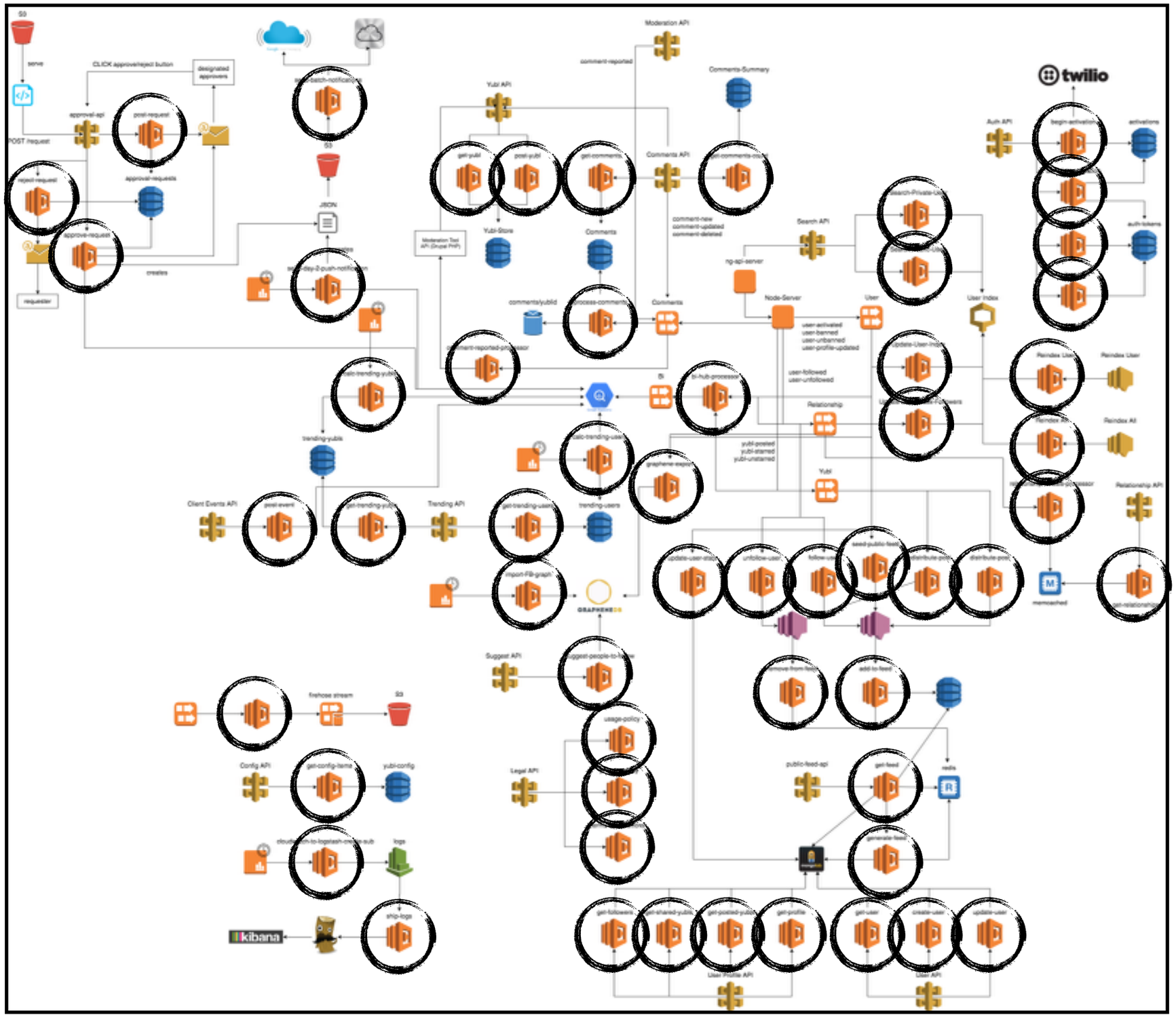
the **risk** of shipping broken software has largely shifted to how your Lambda functions **integrate** with **external services**

serverless

observation #3

smaller units of deployment
means finer grained control of
access, and more things to secure

serverless

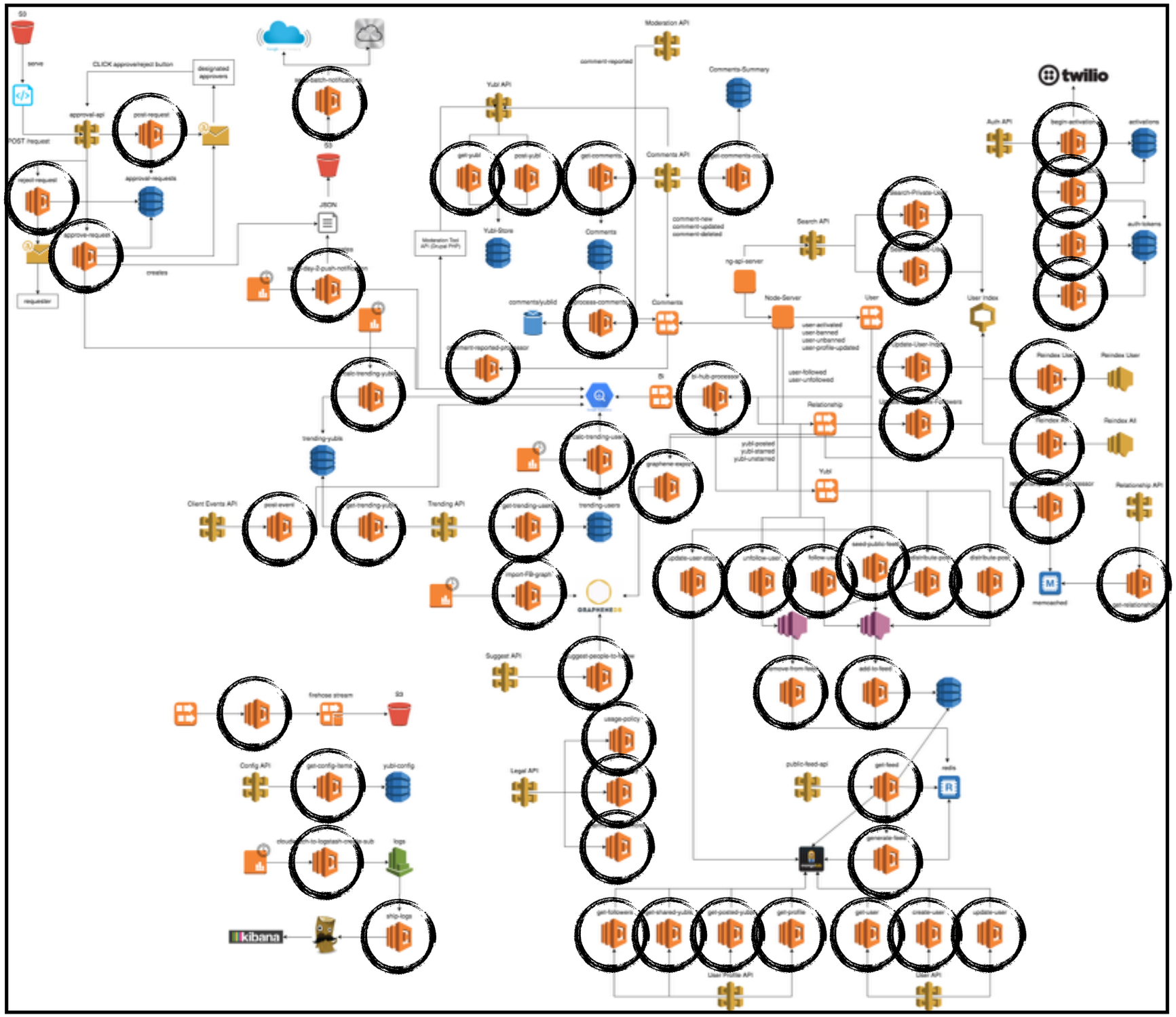

S3
serve
CLICK approve/reject button
designated approvers
POST request
approval-api
post-request
approval-requests
reject-request
approve-request
creates
requester
JSON
Moderation API
comment-reported
Yubl API
Comments-Summary
get-yubl
post-yubl
get-comments
Comments API
Yubl-Store
Comments
Moderation Tool API (Drupal PHP)
comment-new
comment-updated
comment-deleted
Search API
ng-api-server
Node-Server
User
User Index
commentsyubld
process-comments
Comments
user-activated
user-banned
user-unbanned
user-profile-updated
user-followed
user-unfollowed
Relationship
yubl-posted
yubl-starred
yubl-unstarred
Yubl
twilio
Auth API
begin-activation
activations
auth-tokens
Reindex User
Reindex All
Relationship API
memcached
trending-yubls
Client Events API
post-event
get-trending-yubl
Trending API
get-trending-user
trending-users
graphene-export
import-FB-graph
GRAPHENEDB
Suggest API
suggest-people-to-follow
firehose stream
S3
usage-policy
Config API
get-config-items
yubl-config
Legal API
public-feed-api
get-feed
redis
generate-feed
logs
kibana
ship-logs
get-followers
get-shared-yubls
get-posted-yubls
get-profile
get-user
create-user
update-user
User Profile API
User API

serverless

observation #4

smaller units of deployment also means more application configurations in general

serverless


S3
serve
CLICK approve/reject button
designated approvers
POST request
approval-api
post-request
approval-requests
reject-request
approve-request
creates
requester
JSON
Moderation API
comment-reported
Yubl API
Comments-Summary
get-yubl
post-yubl
get-comments
Comments API
Yubl-Store
Comments
Moderation Tool API (Drupal PHP)
comment-new
comment-updated
comment-deleted
Search API
ng-api-server
Node-Server
User
User Index
commentsyubld
process-comments
Comments
user-activated
user-banned
user-unbanned
user-profile-updated
user-followed
user-unfollowed
Relationship
yubl-posted
yubl-starred
yubl-unstarred
Yubl
twilio
Auth API
begin-activation
activations
auth-tokens
Reindex User
Reindex All
Relationship API
memcached
trending-yubls
Client Events API
post-event
get-trending-yubl
Trending API
get-trending-user
trending-users
graphene-export
import-FB-graph
GRAPHENEDB
Suggest API
suggest-people-to-follow
firehose stream
S3
usage-policy
Config API
get-config-items
yubl-config
Legal API
public-feed-api
get-feed
redis
generate-feed
logs
kibana
ship-logs
get-followers
get-shared-yubls
get-posted-yubls
get-profile
get-user
create-user
update-user
User Profile API
User API

serverless

conclusion #2

the **risk** of misconfiguration (both application & IAM) has exploded

serverless

conclusion #3

the **risk** profile for a serverless application is very different to that of a serverful application

test honeycomb