



Dynamic Meta-Level Architecture

Deliverable 1.1.2

Laurent Christophe, Coen De Roover, Elisa Gonzalez Boix

tierless programming



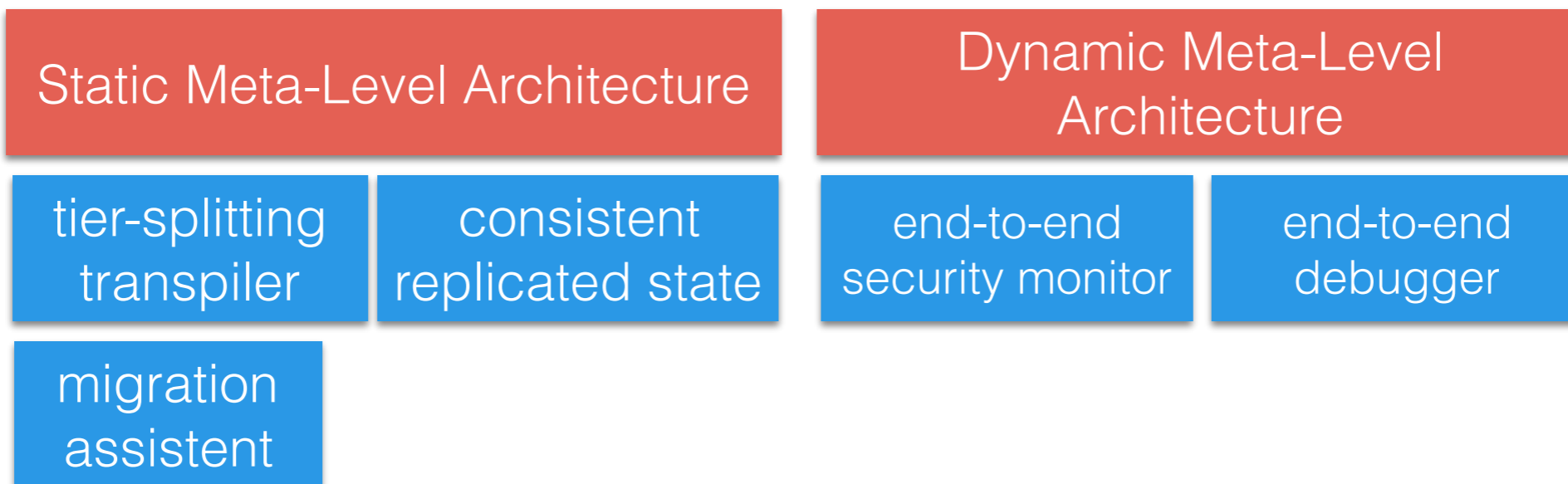
enabling technologies




libraries



tools



Aran: Instrumenting JavaScript

 <https://www.npmjs.com/aran>

1000+ download/month

Program Instrumentation

```
function fac (n) {  
  if (n === 0)  
    return 1;  
  return n * fac(n-1);  
}  
fac(1000);
```

```
function fac (n) {  
  if (n === 0)  
    return 1;  
  return n * fac(n-1);  
}  
var time = performance.now();  
fac(1000);  
console.log(performance.now() - time);
```

end-to-end
security monitor

end-to-end
debugger

Dynamic
Meta-Level
Architecture

- 1) Program
instrumentation
- 2) Runtime
instrumentation



Aran's Instrumentation

```
function fac (n) {  
  if (n === 0)  
    return 1;  
  return n * fac(n-1);  
}  
fac(6);
```

target.js

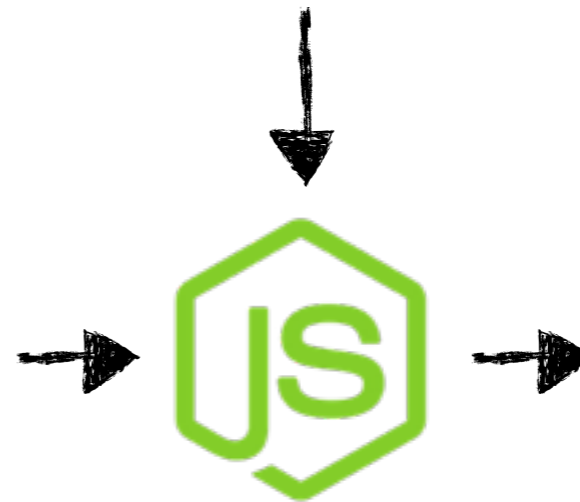
> aran --namespace meta --traps apply

```
function fac (n) {  
  if (n === 0)  
    return 1;  
  return n * meta.apply(fac, null, [n-1], 11);  
}  
meta.apply(fac, null, [6], 17);
```

instrumented.js

```
var d = "";  
var meta = {};  
meta.apply = function (f, t, xs, i) {  
  console.log(d+f.name+"("+xs+")@"+i);  
  d += ".";  
  var r = f.apply(t, xs);  
  d = d.substring(1);  
  console.log(d+r);  
  return r;  
};
```

analysis.js



```
fac(4)@17  
  . fac(3)@11  
    .. fac(2)@11  
      ... fac(1)@11  
        .... fac(0)@11  
          ..... 1  
            ..... 1  
              .. 2  
                . 6  
                  24
```

Aran's Capabilities

```
1 var traps = {};  
2 // General //  
3 traps.Program = function (idx) { };  
4 traps.Strict = function (idx) { };  
5 // Creation //  
6 traps.primitive = function (prm, idx) { return prm };  
7 traps.closure = function (fct, idx) { return fct };  
8 traps.object = function (prps, idx) {  
9   var obj = {};  
10  for (var i=0; i<prps.length; i++)  
11    Object.defineProperty(obj, prps[i].key, prps[i]);  
12  return obj;  
13 };  
14 traps.array = function (vals, idx) { return vals };  
15 traps.regexp = function (ptn, flg, idx) { return new RegExp(ptn, flg) };  
16 // Environment //  
17 traps.Declare = function (kind, tags, idx) { };  
18 traps.read = function (tag, val, idx) { return val };  
19 traps.write = function (tag, val, wrt, idx) { return wrt(val) };  
20 traps.Enter = function (idx) { };  
21 traps.Leave = function (idx) { };  
22 traps.with = function (env, idx) { return env };  
23 // Apply //  
24 traps.apply = function (fct, ths, args, idx) { return fct.apply(ths, args) };  
25 traps.construct = function (cst, args, idx) { return new cst(...args) };  
26 traps.Arguments = function (args, i) { };  
27 traps.return = function (val, idx) { return val };  
28 traps.eval = function (args, idx) { return args[0] };  
29 traps.unary = function (opr, arg, idx) { return eval(opr+" arg") };  
30 traps.binary = function (opr, lft, rgt, idx) { return eval("lft "+opr+" rgt") };  
31 // Object //  
32 traps.get = function (obj, key, idx) { return obj[key] };  
33 traps.set = function (obj, key, val, idx) { return obj[key] = val };  
34 traps.delete = function (obj, key, idx) { return delete obj[key] };  
35 traps.enumerate = function (obj, idx) {  
36   var keys = [];  
37   for (var key in obj)  
38     keys.push(key);  
39   return keys;  
40 };  
41 // Control //  
42 traps.test = function (val, idx) { return val };  
43 traps.Label = function (lbl, idx) { };  
44 traps.Break = function (lbl, idx) { };  
45 traps.throw = function (err, idx) { return err };  
46 traps.Try = function (idx) { };  
47 traps.catch = function (err, idx) { return err };  
48 traps.Finally = function (idx) { };  
49 traps.sequence = function (vals, idx) { return vals };  
50 traps.expression = function (val, idx) { return val };  
51 // Exports //  
52 global._meta_ = traps;
```

compat
ES5

- 1) Control flow inspection
- 2) Environment inspection
- 3) Objects profiling
- 4) Function profiling
- 5) Time complexity
- 6) Space complexity


Dynamic code
evaluation

```
eval("fac(4)");
```

Multiple runtimes



Demonstration

 <https://www.npmjs.com/otiluke>

500+ downloads/month

Linvail: Tracking JavaScript Values



Linvail: A General-Purpose Platform for Shadow Execution of JavaScript
Laurent Christophe, Elisa Gonzalez Boix, Wolfgang De Meuter, Coen De Roover
2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)



<https://www.npmjs.com/linvail>

Case study: Bug Diagnosis

undefined  "#bdate.avlue"

```
var year = document.getElementById("bdate").avlue;  
alert("Your age is: " + (2016 - year));
```

Your age is: NaN

OK

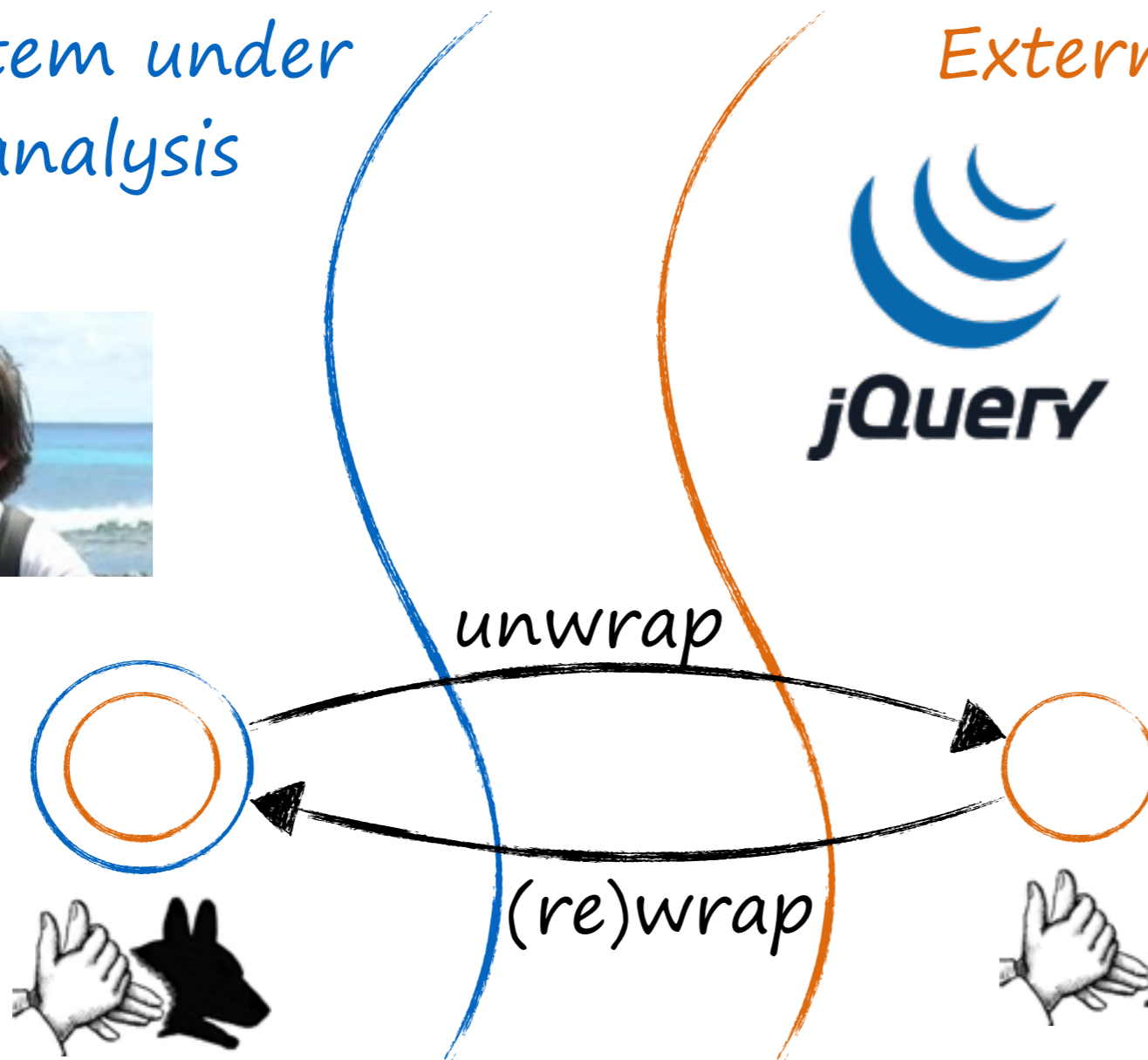
NaN  "2016 - (#bdate.avlue::undefined)"

Membranes to the Rescue

```
var wrapper = {concrete:NaN, meta:"bdate.avlue::undefined"};
```

System under analysis

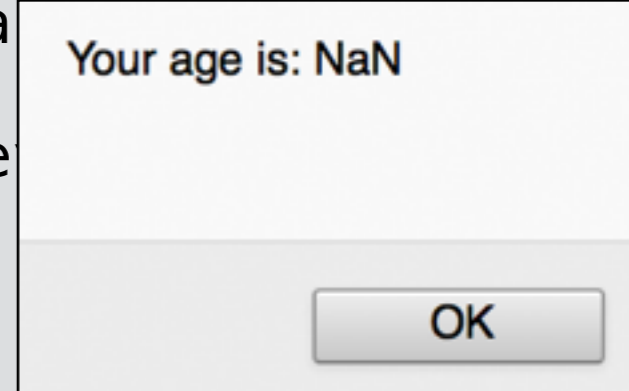
External world



Linvail's API

```
var Aran = require("aran");
var Linvail = require("linvail");
module.exports = function (options) {
  var ctx = null;
  function unwrap () {
    return this.meta["get", #bdate, "avalue", 3]
  }
  var linvail = Linvail(function (value) {
    if (!value)
      return null;
    return value;
  });
  global.meta = Object.assign({}, linvail);
  global.meta.get = function () {
    ctx = [].concat.apply("get", arguments);
    return linvail.get.apply(null, arguments);
  };
  global.meta["binary", "-", 2016, {
    concrete: undefined,
    meta: ["get", #bdate, "avalue", 3]
  }];
  return aran.instrument;
};
```

```
var year = document.getElementById("bdate").avalue;
alert("Your age is: " + (2016 - year));
```

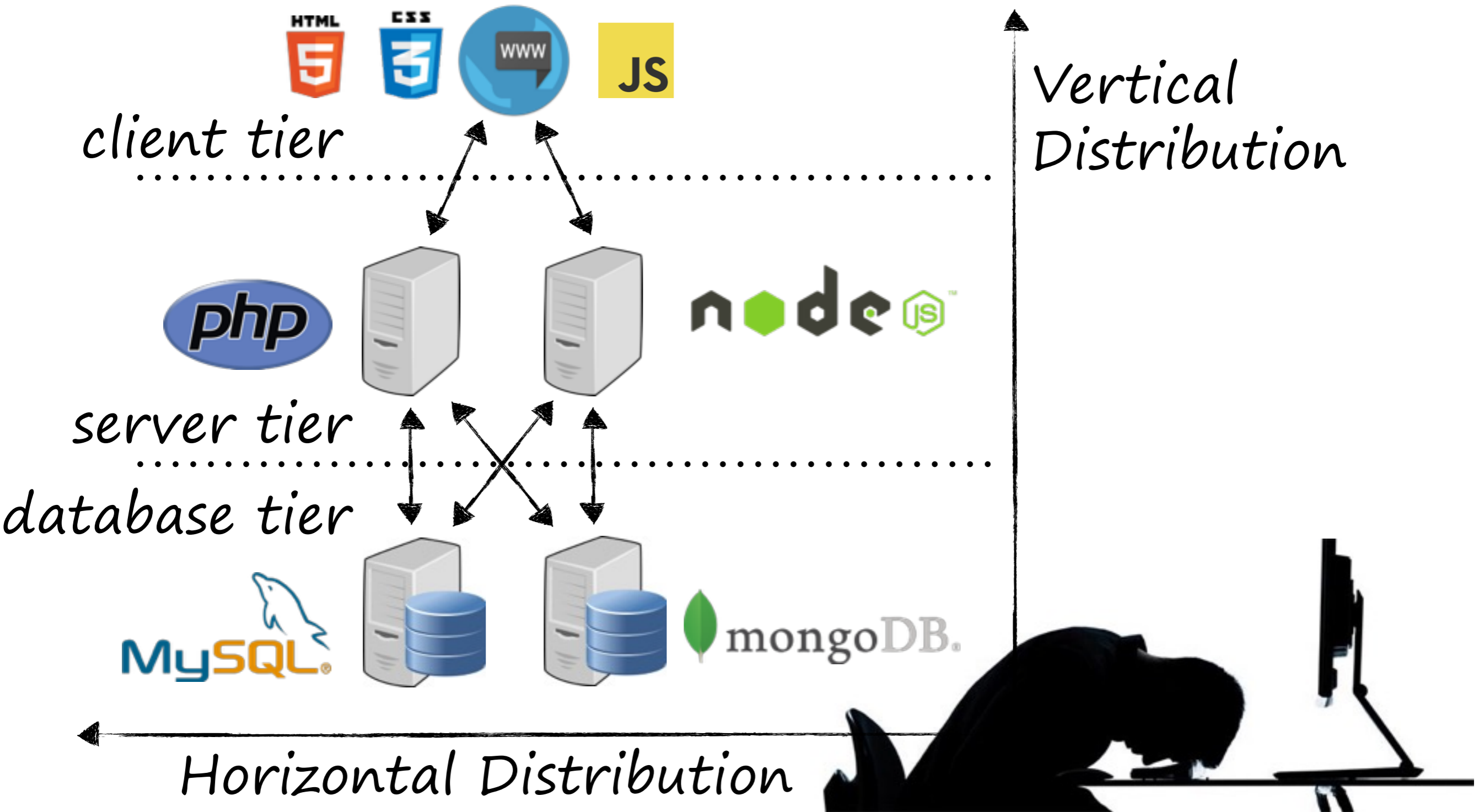


Linvail's Applicability

Benchmark	LoC	LoC*	Identity	Track-NaN	Track-Void	Taint	Constraint
3d-cube	357	815	336	396	386	376	365
3d-morph	65	36	3719	3856	4013	4286	4236
3d-raytrace	449	704	556	741	720	730	710
access-binary-trees	56	70	1739	3086	2180	1958	1786
access-fannkuch	72	72	11681	16068	1474	13547	9436
access-nbody	176	145	9202	14502	9249	10985	7045
access-nsieve	48	51	4278	6924	6910	5190	3705
bitops-3bit-bits-in-byte	42	29	10677	29102	18902	17654	13346
bitops-bits-in-byte	32	31	11122	19895	17802	12938	12855
bitops-bitwise-and	37	11	4547	16267	10932	7031	6303
bitops-nsieve-bits	44	45	15413	14581	14201	32	13534
controlflow-recursive	34	72	5605	2826	7201	6391	5700
crypto-aes	428	735	2199	1820	2521	2990	2266
crypto-md5	294	830	2289	2128	2819	2911	1785
crypto-sha1	230	248	2617	3471	3185	4500	2532
date-format-tofte	303	379	1851	2036	2015	2243	1959
date-format-xparb	422	829	43	101	183	48	43
math-cordic	108	79	12436	15227	15401	17350	13376
math-partial-sums	47	44	474	577	601	693	462
math-spectral-norm	61	81	10087	10162	14891	14186	10865
regexp-dna	1723	105	5	7	4	5	5
string-base64	139	199	2022	1492	1720	2465	2029
string-fasta	92	189	1674	1931	2184	2623	2628
string-tagcloud	271	347	18	18	12	22	21
string-unpack-code	83	185	75	29	45	74	44
string-validate-input	94	74	87	355	286	217	251
Average	–	–	4444	6446	5378	5055	4511
First Quartile	–	–	897	396	386	10985	365
Median	–	–	3719	3856	4013	4286	2628
Third Quartile	–	–	10087	10162	7201	2623	6303

Tearless Dynamic Meta-level Architecture

Web Programming



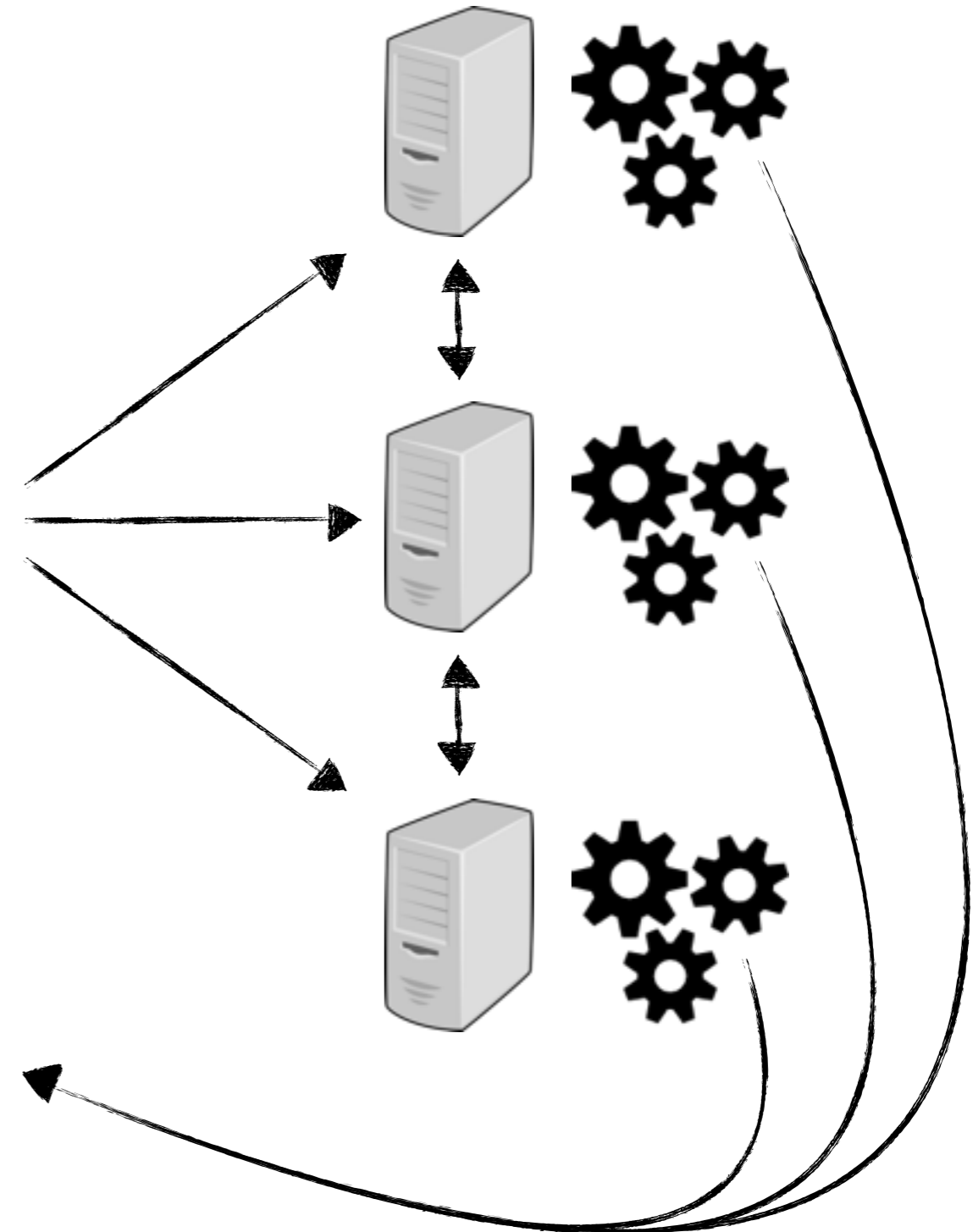
Tierless Dynamic Tools



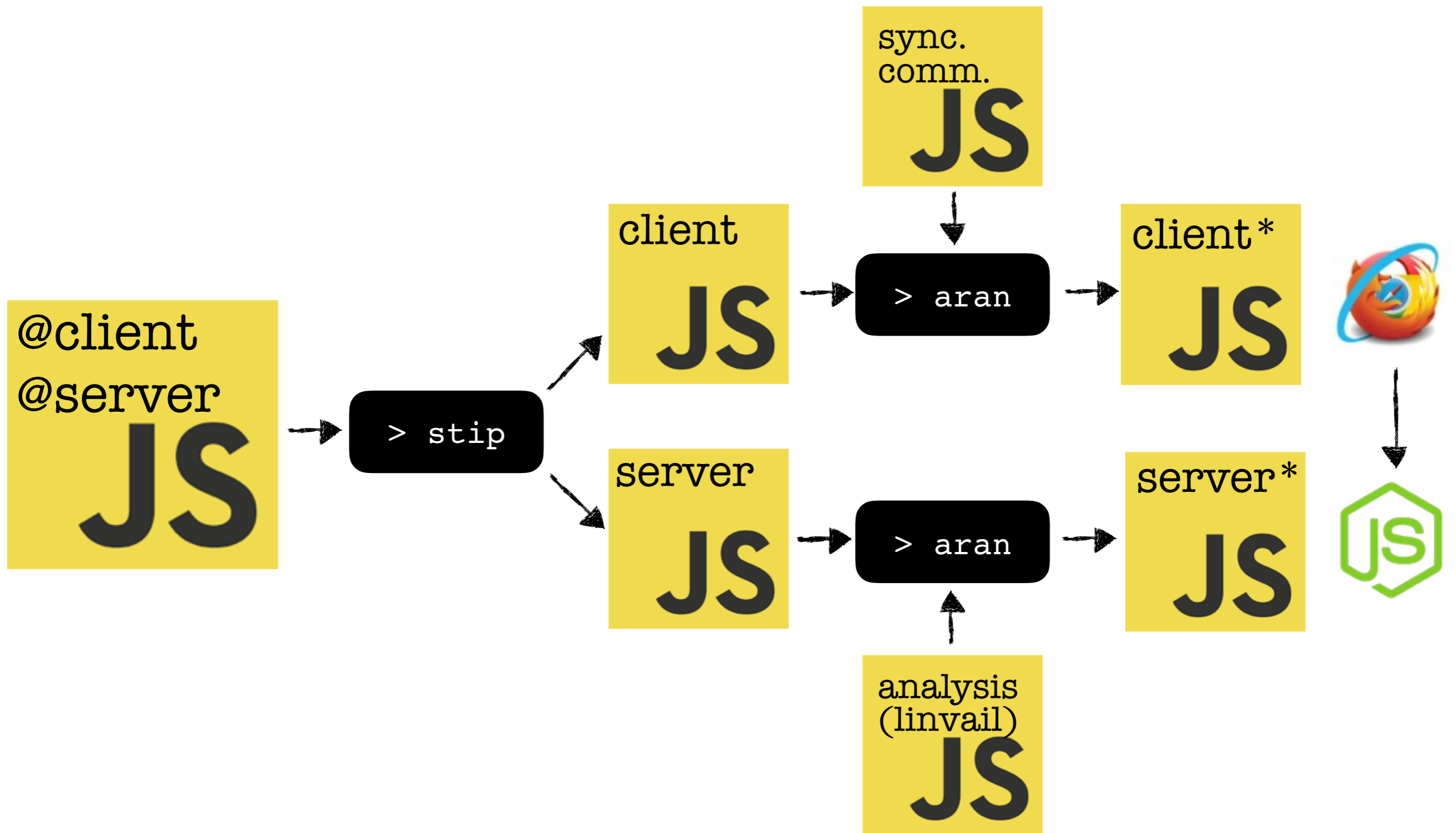
@tierA
@tierB
@tierC

```
for i in people.data.users:  
    response = client.api.statuses.user_timeline.get(screen_name=i.scre  
    print 'Got', len(response.data), 'tweets from', i.screen_name  
    if len(response.data) != 0:  
        ldate = response.data[0]['created_at']  
        ldate2 = datetime.strptime(ldate, '%a %b %d %H:%M:%S +0000 %Y'  
        today = datetime.now()  
        howlong = (today - ldate2).days  
        if howlong < daywindow:  
            print i.screen_name, 'has tweeted in the past', daywindow,  
                totaltweets += len(response.data)  
            for j in response.data:  
                if j.entities.urls:  
                    for k in j.entities.urls:  
                        newurl = k['expanded_url']  
                        urliset.add(newurl, j.user.screen_name)  
        else:  
            print i.screen_name, 'has not tweeted in the past', daywind
```

```
Apply push@//www.google.be:443/search?  
q=asdd&ie=utf-8&oe=utf-8&client=firefox-b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:283  
Apply getElementById@//www.google.be:443/  
search?q=asdd&ie=utf-8&oe=utf-8&client=firefox-  
b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:282  
Apply querySelector@//www.google.be:443/search?  
q=asdd&ie=utf-8&oe=utf-8&client=firefox-b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:282  
Apply @//www.google.be:443/search?  
q=asdd&ie=utf-8&oe=utf-8&client=firefox-b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:282  
Apply remove@//www.google.be:443/search?  
q=asdd&ie=utf-8&oe=utf-8&client=firefox-b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:191  
Apply @//www.google.be:443/search?  
q=asdd&ie=utf-8&oe=utf-8&client=firefox-b-  
ab&gfe_rd=cr&ei=hvo3WJvyEc7G8AfmkJ8w#171286:282  
Apply remove@//www.google.be:443/search?
```



Cross-tier Analysis



Tech Stack

