Performance Troubleshooting with SharePoint Online and Office 365

@spmcdonough on Twitter (for heckling purposes)

Sean P. McDonough
Microsoft MVP (Office Development, Office Servers and Services)
Chief Technology Officer and Owner
Bitstream Foundry LLC

bitstream FOUNDRY

MVP Microsoft Most Valuable Professional

Some of the things we do at

bitstream FOUNDRY

- Performance troubleshooting
- Farm design and build
- Online tenant preparations
- On-prem and cloud migrations
- Full-trust custom development
- Add-on based development

Today's Agenda: Part 1

Focus on SPO

- Introduction
- SharePoint Online (SPO) Implementation
- SharePoint Online Diagnostics and Tools
- Design and Development Guidance
- Samples, Examples, and Demos

Today's Agenda: Part 2

Focus on Network

- The Reality of Plumbing
- Talking to SPO: What's Involved?
- A Few Words on Capacity
- Troubleshooting and Associated Tools
- Samples, Examples, and Demos
- References

Remember ...

This is a workshop, and over the next several hours I assume that you'll have questions about the concepts I'm presenting, the tools I'm demonstrating, and the troubleshooting strategies I use ...

Please speak up!

If you have a question, chances are someone else is wondering

strategies I use ...

Please speak up!

If you have a question, chances are someone else is wondering too! I'm here to help, and if I don't have an answer, I can track one down for you later.

But first ...


An important note

This is
Office 365

hanging and updating it"

Please don

An important note

Office 365 is an "evergreen service"

eaning "Microsoft is always changing and updating

An important note

Office 365 is an "evergreen service"

meaning "Microsoft is always changing and updating it"

What I show you today ...

- will probably be true tomorrow

always changing and updating

What I show you today ...

- will probably be true tomorrow
- has a good chance of being true next week
- might be true in month
- probably worth questioning and re-evaluating in a year

Please don't dig this up in five years and then send me hate mail because I presented something that is no longer accurate due to a SharePoint Online service change.

Please don't dig this up in five years and then send me hate mail because I presented something that is no longer accurate due to a SharePoint Online service change.

Dear Sean,

I was reviewing a workshop you put together five years ago, and I found elements that were incorrect. You are a horrible person and you should never touch SharePoint Online again.

Love you lots!

- an attendee

First Stop:

Some basic SharePoint farm architecture

(and why that matters with SPO)


Welcome to the farm!

might think that SPO is simply an extension of this patt

On-premises SharePoint farms come in all shapes and sizes

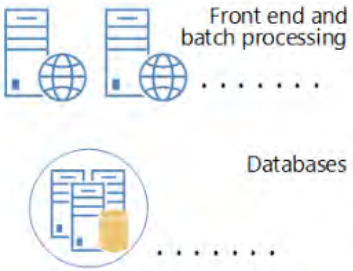
Small Farm

<1,000 users

Fault tolerance for simple workloads with small volumes of content

Two tiers:
- Combined front-end and batch processing servers
- Database servers

Scale the number of servers as needed.



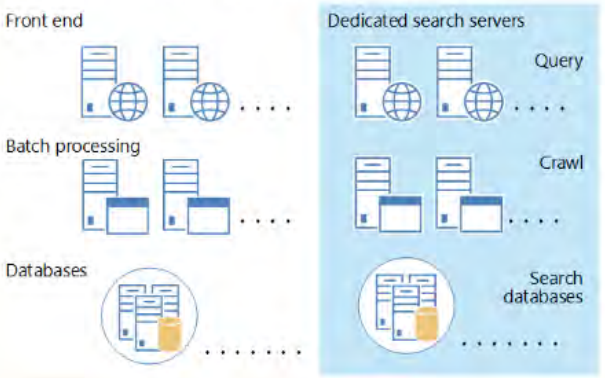
Medium Farm

<10,000 users

Dedicated search servers for up to 10 million items.

Three tiers:
- Front-end servers
- Batch processing servers
- Database servers

Scale the number of servers as needed.

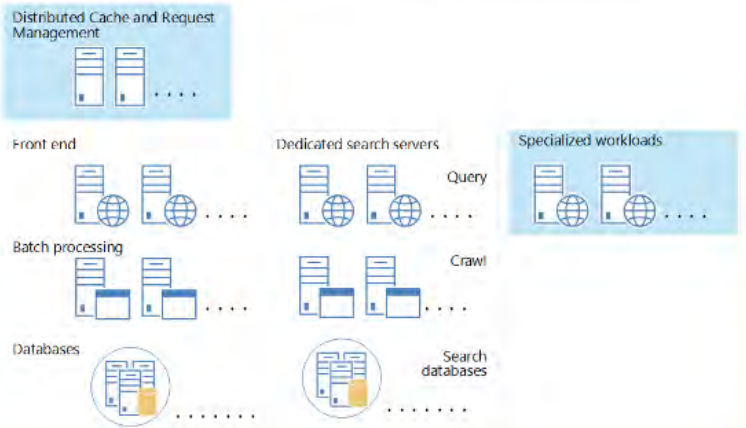


Large Farm

More than 10,000 users

Additional server types to support large farms.

This farm represents each of the server roles that are recommended. For each server role the servers are configured identically. Scale each server role independently. Large farms benefit by adding dedicated servers for Distributed Cache and by adding Request Management.



well, not really ...

You might think that SPO is simply an extension of this pattern.

On-premises SharePoint farms come in all shapes and sizes

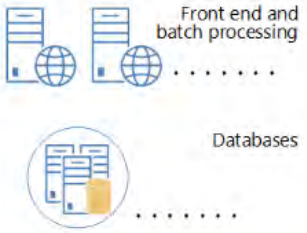
Small Farm

<1,000 users

Fault tolerance for simple workloads with small volumes of content

Two tiers:
- Combined front-end and batch processing servers
- Database servers

Scale the number of servers as needed.



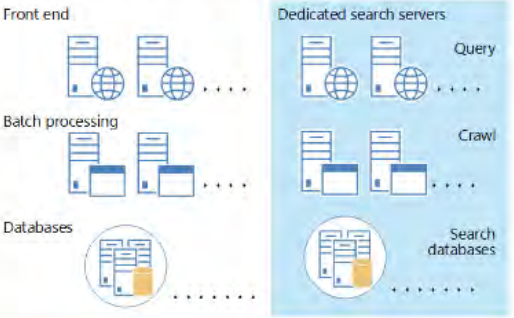
Medium Farm

<10,000 users

Dedicated search servers for up to 10 million items.

Three tiers:
- Front-end servers
- Batch processing servers
- Database servers

Scale the number of servers as needed.

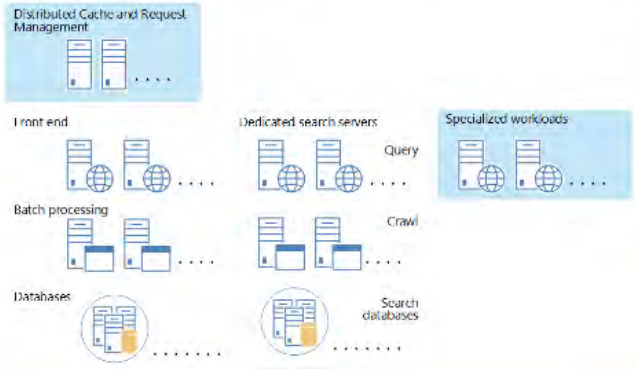


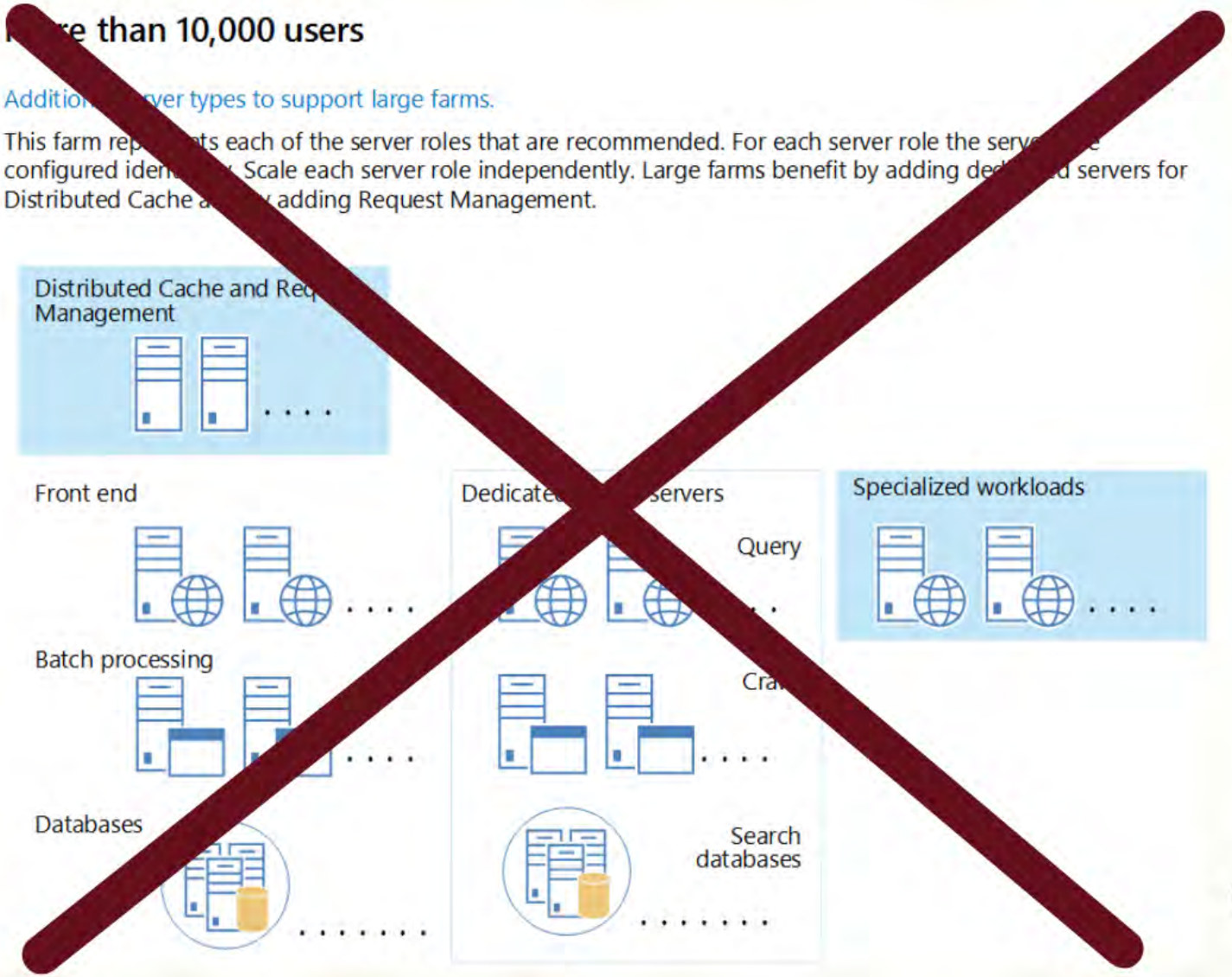
Large Farm

More than 10,000 users

Additional server types to support large farms.

This farm represents each of the server roles that are recommended. For each server role the servers are configured identically. Scale each server role independently. Large farms benefit by adding dedicated servers for Distributed Cache and by adding Request Management.





It is not.

well, not really ...

You might think that SPO is simply an extension of this pattern.

This is a stamp
too (well, several of them)

This is a stamp too (well, several of them)

Datacenter 1..N:

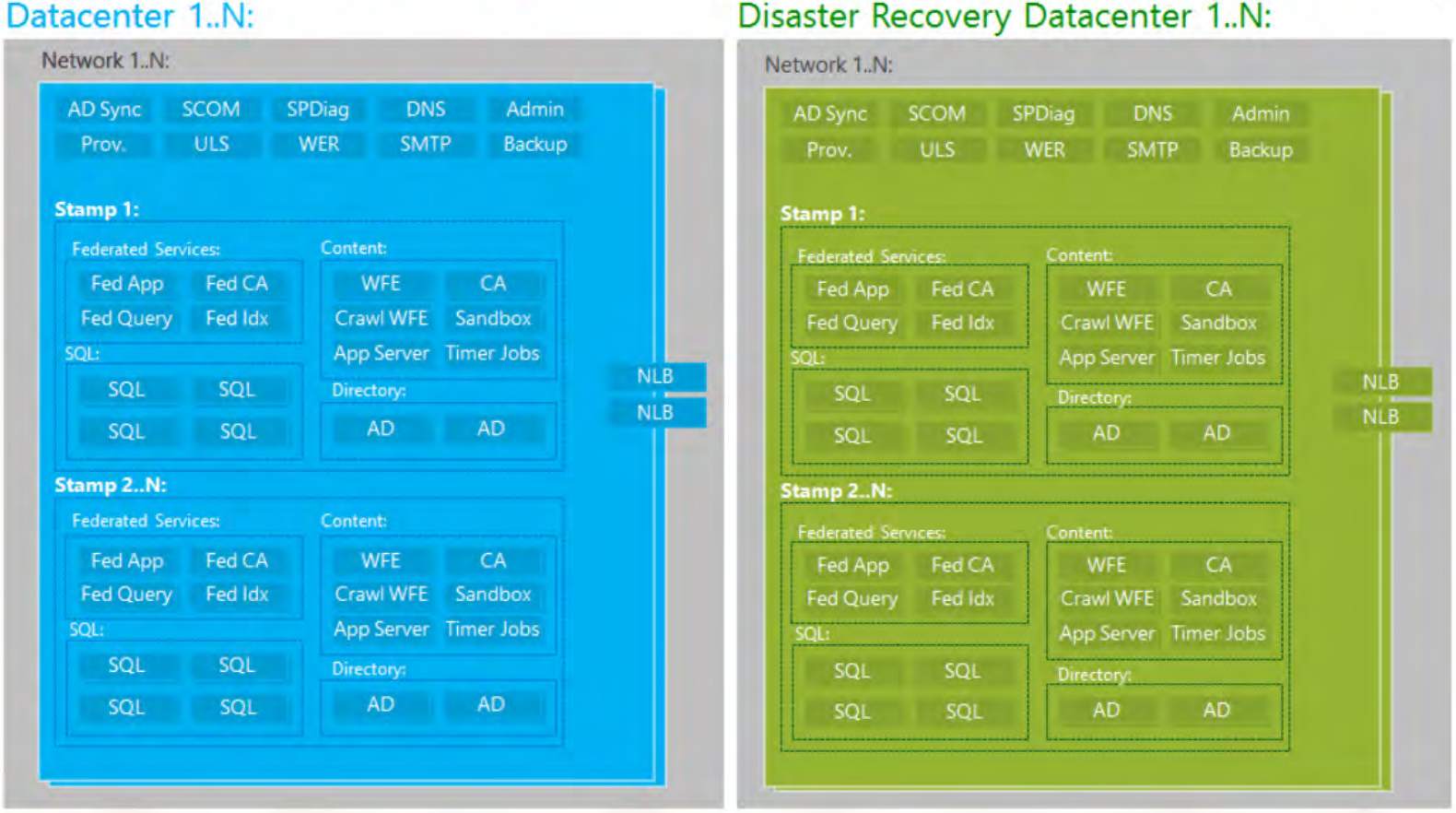
Network 1..N:

Disaster Recovery Datacenter 1..N:

Network 1..N:

Grid Manager

This is a stamp too (well, several of them)

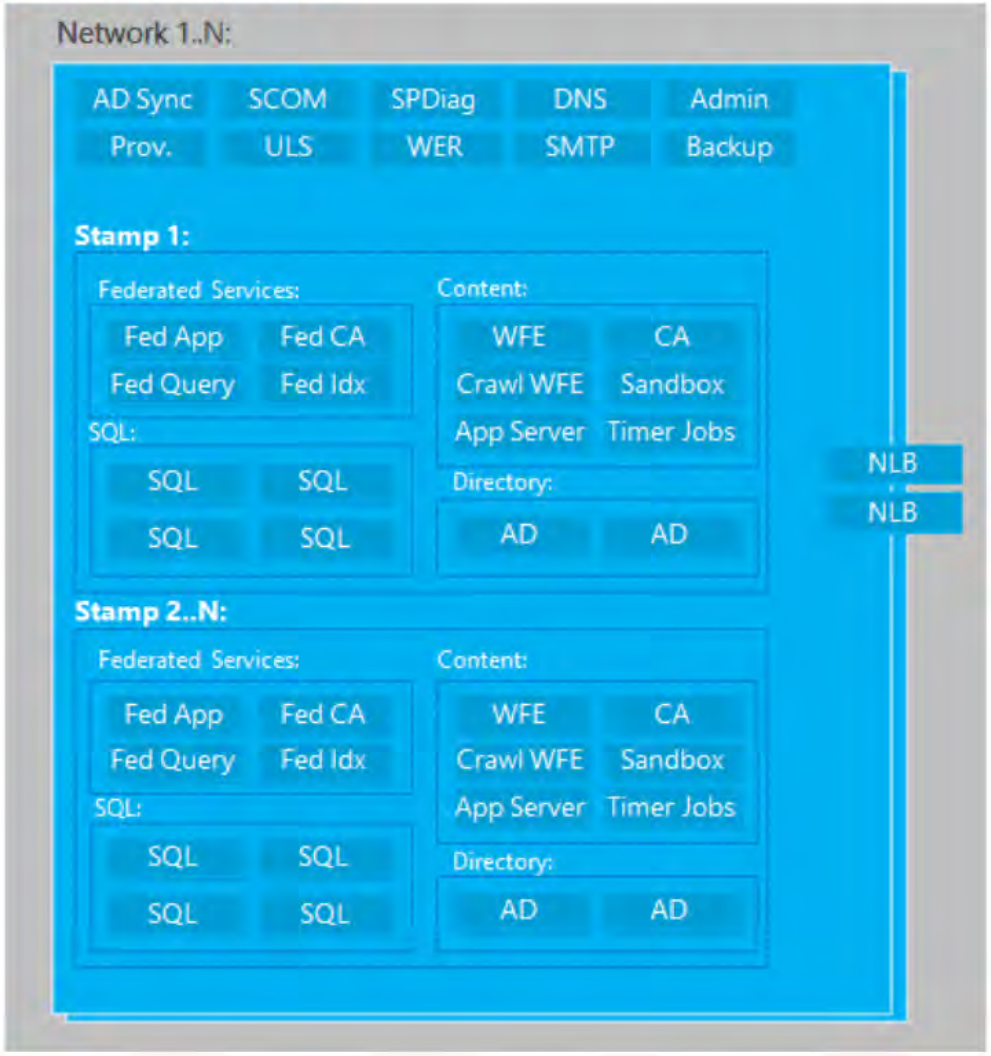


- Grid Manager
- Global Directory
- Tenant Admin (UI)
- Commerce backend
- DNS (multiple)
- OrgID Auth, Svc.
- Incident Management
- Azure (Windows/SQL)
- CDN Services

Looking at the representation of an individual stamp, you might think it's only 16 servers.

This
too (well,

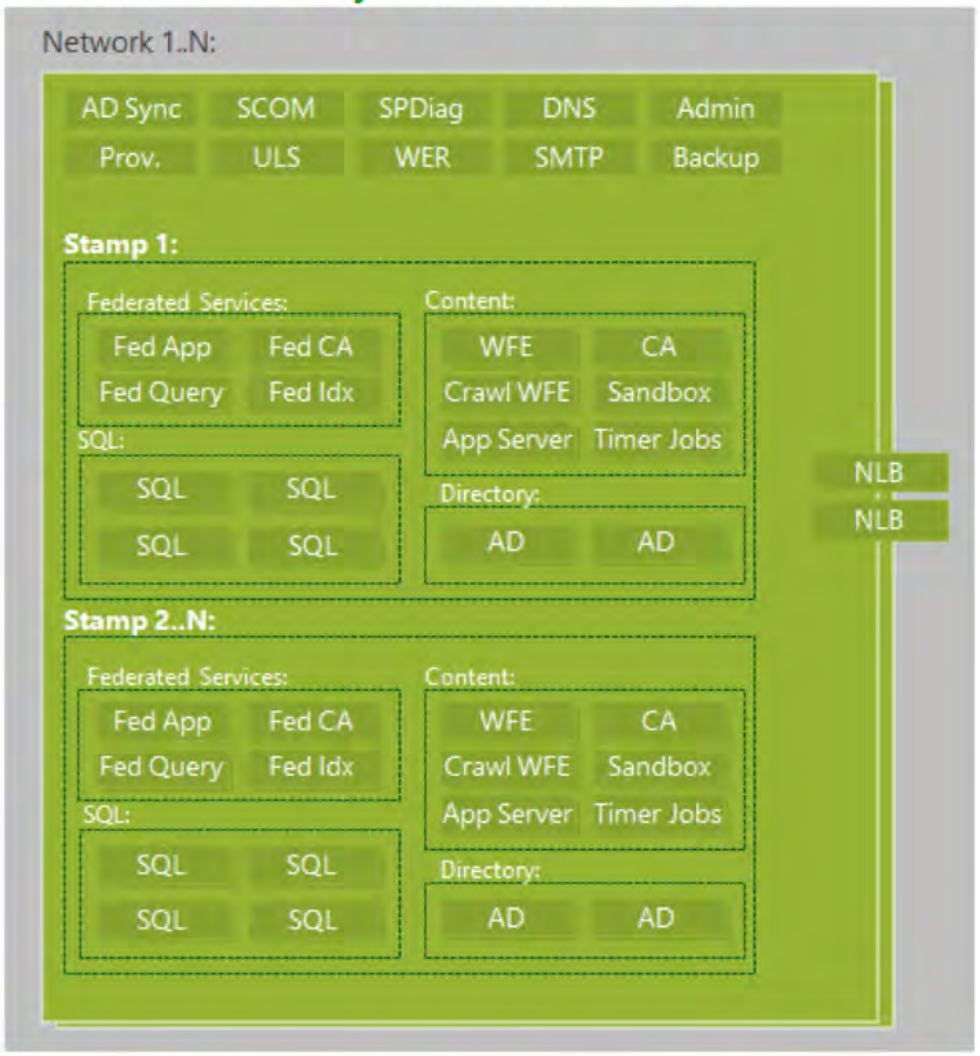
Each datacenter has two or more stamps per SPO environment for high-availability.



Looking at the representation of an individual stamp, you might think it's only 16 servers.

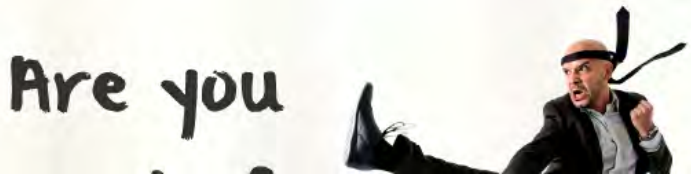
s is a stamp
several of them)

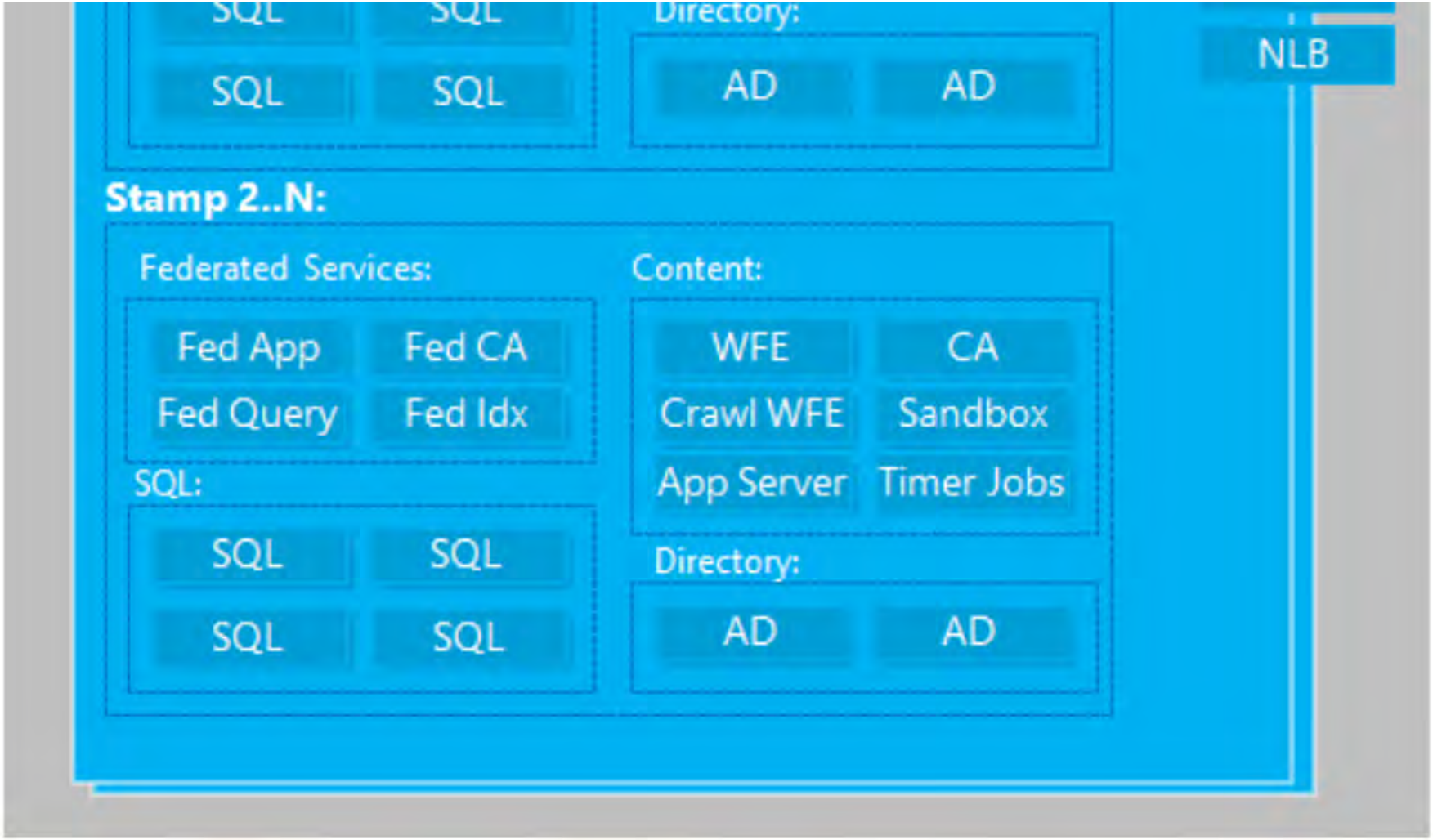
Disaster Recovery Datacenter 1..N:



- Grid Manager
- Global Directory
- Tenant Admin (UI)
- Commerce backend
- DNS (multiple)
- OrgID Auth, Svc.
- Incident Management
- Azure (Windows/SQL)
- CDN Services

Additional stamps exist in a different region for redundancy and failover.



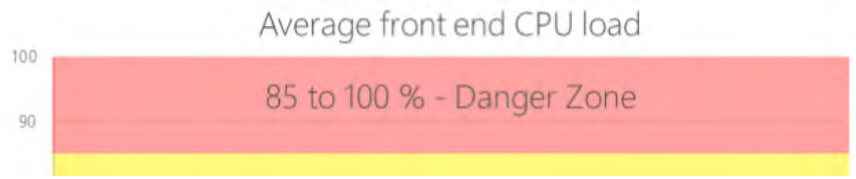


Looking at the representation of an individual stamp, you might think it's only 16 servers.

Are you ready for the kicker?

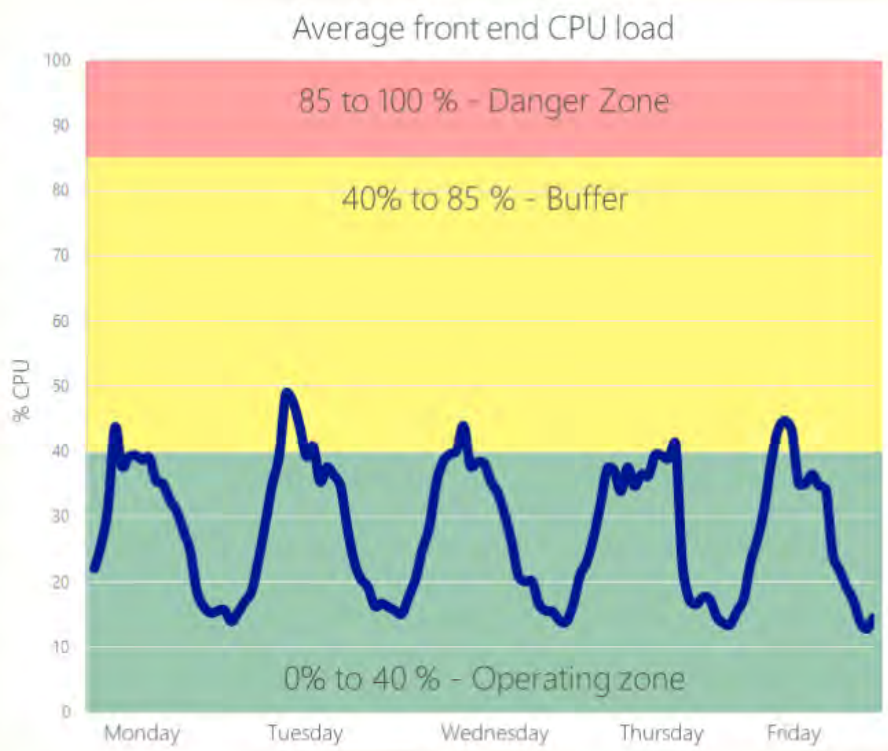
The exact number of servers in a SharePoint Online stamp is variable.

The number of servers per



The exact number of servers in a SharePoint Online stamp is variable.

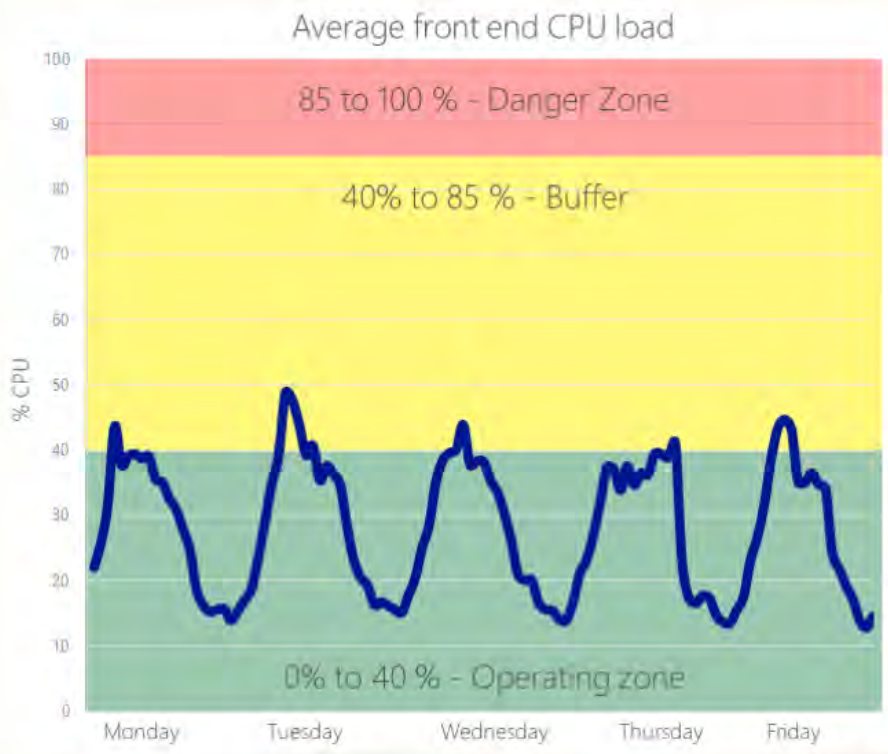
The number of servers per stamp varies because the server count is adjusted based on average front-end CPU load.



- If load rises above 40%, additional servers are automatically provisioned and added to the stamp

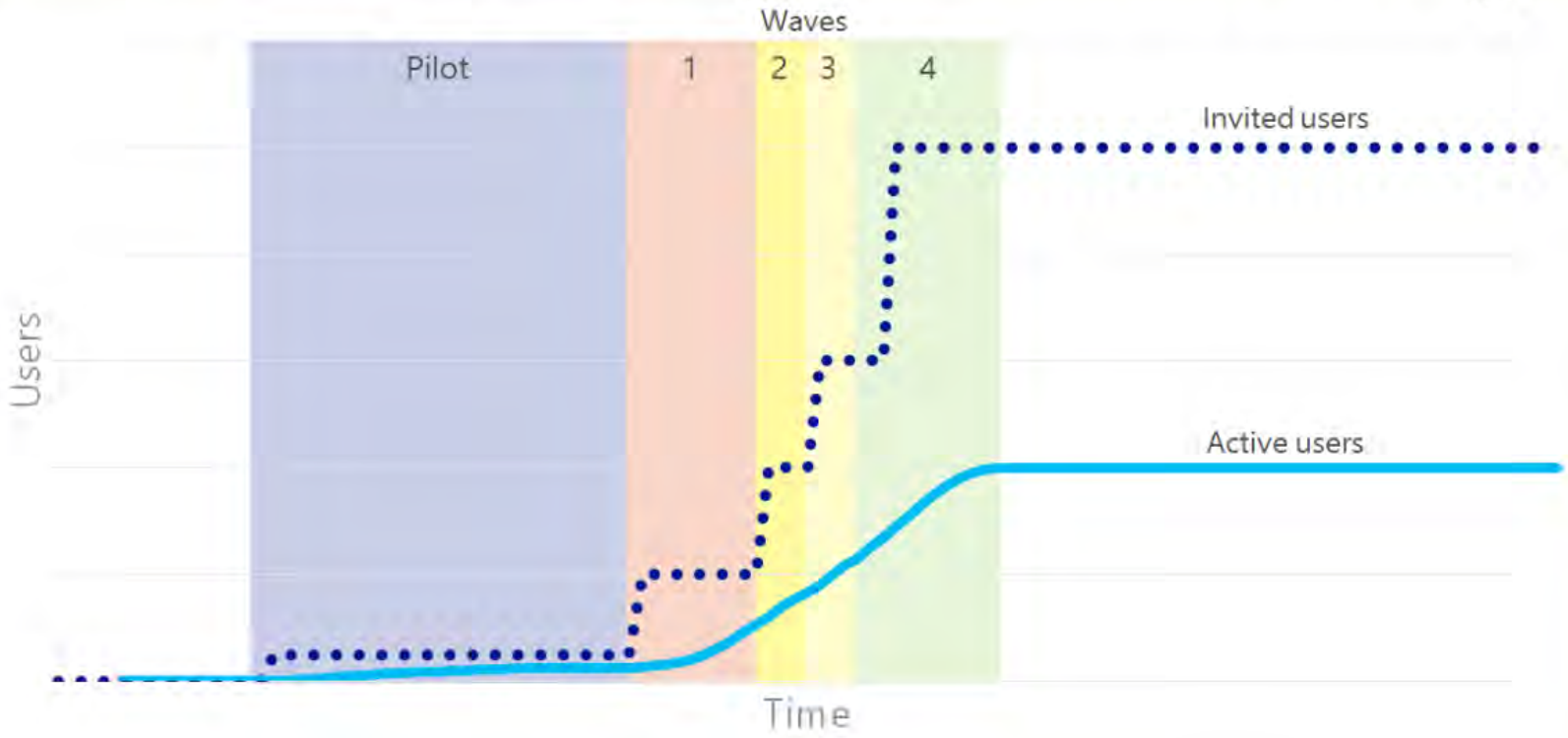
SharePoint Online stamp is variable.

The number of servers per stamp varies because the server count is adjusted based on average front-end CPU load.



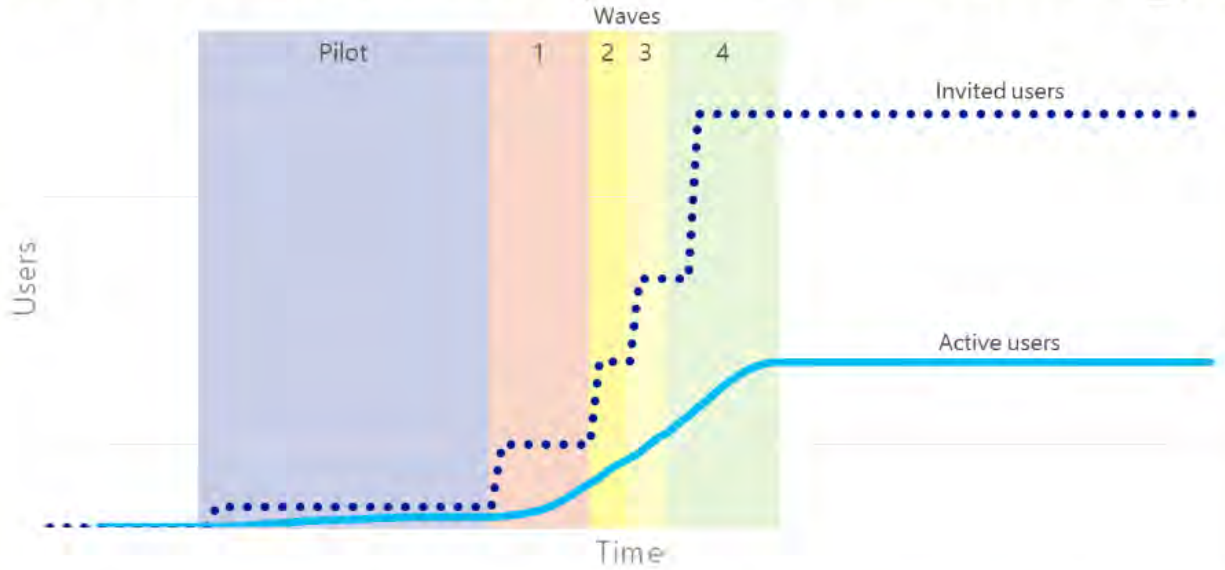
- If load rises above 40%, additional servers are automatically provisioned and added to the stamp.
- If load drops, servers are decommissioned.

Adding and removing is not an instantaneous process, though.



an control load (for example, gradually increasing the nu
users over time when you roll out a site), you should. It w

Adding and removing is not an instantaneous process, though.



If you can control load (for example, gradually increasing the number of invited users over time when you roll out a site), you should. It will give the provisioning system time to adjust/compensate for growing load.

If you remember only one thing in this discussion of stamps and elastic capacity, please let it be this one point ...

Load testing is futile.

astic nature of a stamp, there's really no way to effectively load t

please let it be this one point ...

g the number of
uld. It will give
growing load.

Load testing is futile.

- Given the elastic nature of a stamp, there's really no way to effectively load test SPO. Any numbers you get or produce are essentially meaningless in the grand scheme of things.
- Rather than load testing, focus instead on the items we're going to cover in the rest of this presentation. They'll help you avoid poorly performing pages and sites.


Welcome to the farm!

Okay, so it really feels like there is a tangible performance problem.

Okay, so it really feels like there is a tangible performance problem.

How do you prove it objectively?

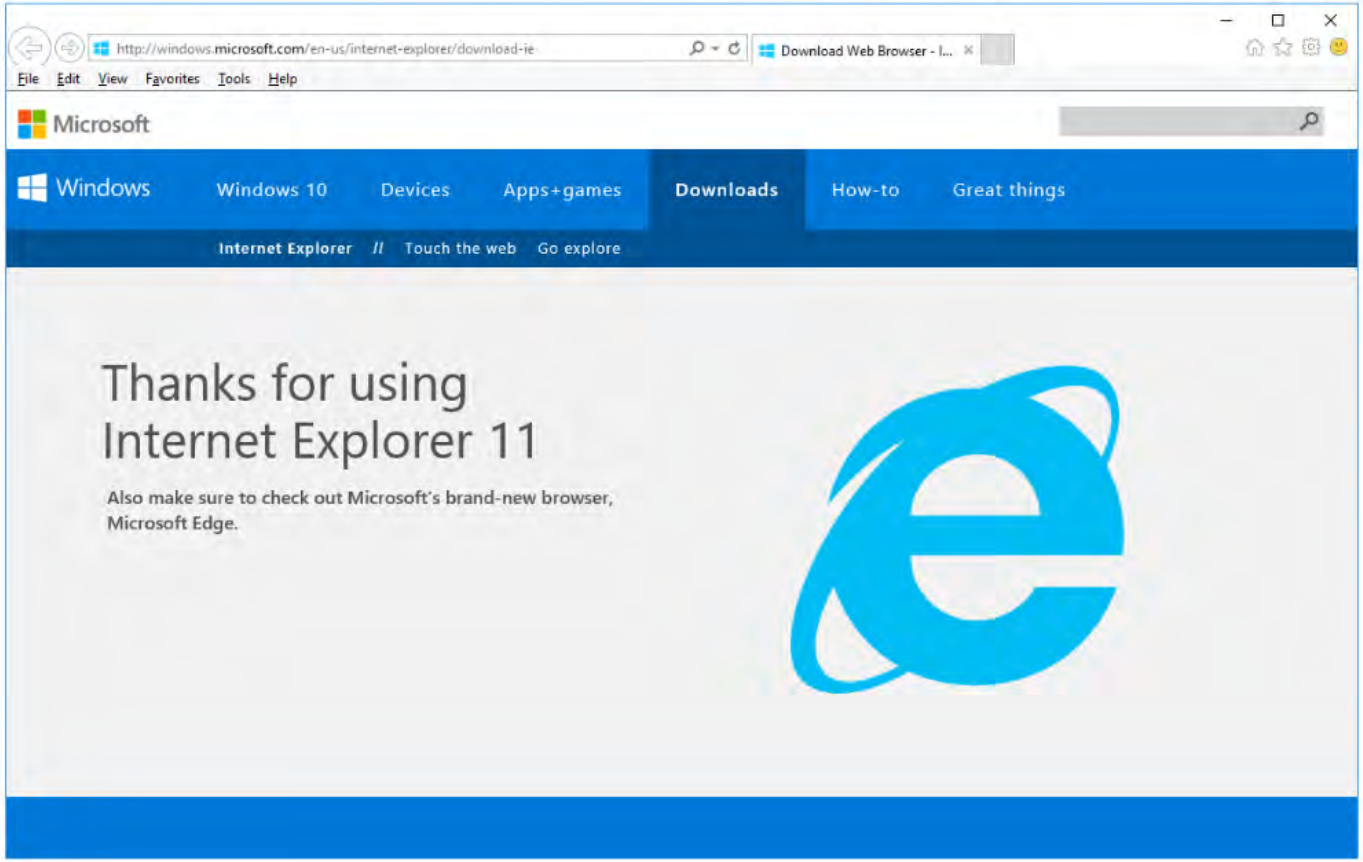
The good news: the destruction of computer equipment is not necessary (although you might feel compelled to do so)!

objectively?

Meet your primary diagnostic tool.

Meet your primary diagnostic tool.

In all likelihood, you already have it on your system.



Hello, Internet Explorer!

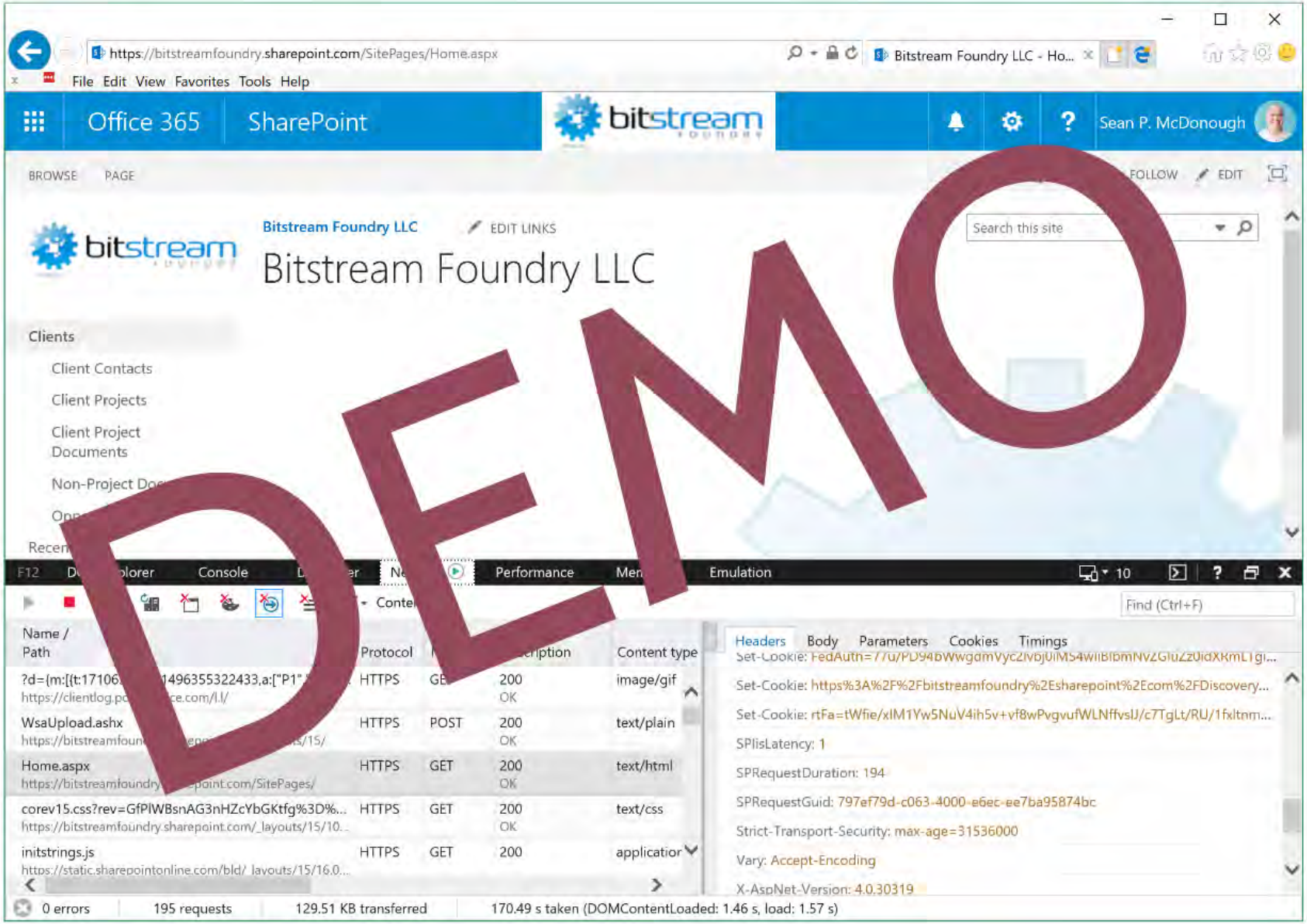
Un

Um ... you're kidding, right?

Nope!

Um ... you're kidding, right?

Nope!


https://bitstreamfoundry.sharepoint.com/SitePages/Home.aspx
Bitstream Foundry LLC - Ho...
File Edit View Favorites Tools Help
Office 365
SharePoint
bitstream
Sean P. McDonough
BROWSE
PAGE
FOLLOW
EDIT
Bitstream Foundry LLC
EDIT LINKS
Search this site
Clients
Client Contacts
Client Projects
Client Project Documents
Console
Performance
Emulation
Find (Ctrl+F)
Name / Path
Protocol
Content type
Headers
Body
Parameters
Cookies
Timings
Set-Cookie: https%3A%2F%2Fbitstreamfoundry%2Esharepoint%2Ecom%2FDiscovery...
Set-Cookie: rtFa=tWfie/xIM1Yw5NuV4ih5v+vf8wPvgvufWLNffvsIJ/c7TgLt/RU/1fxltnm...
SPIisLatency: 1
SPRequestDuration: 194
SPRequestGuid: 797ef79d-c063-4000-e6ec-ee7ba95874bc
Strict-Transport-Security: max-age=31536000
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
?d=(m:[(t:17106
https://clientlog.po
ce.com/l.l/
HTTPS
GE
200
OK
image/gif
WsaUpload.ashx
https://bitstreamfoun
HTTPS
POST
200
OK
text/plain
Home.aspx
https://bitstreamfoundry
epoint.com/SitePages/
HTTPS
GET
200
OK
text/html
corev15.css?rev=GfPlWBsnAG3nHZcYbGKtfg%3D%...
https://bitstreamfoundry.sharepoint.com/_layouts/15/10...
HTTPS
GET
200
OK
text/css
initstrings.js
https://static.sharepointonline.com/bld/_layouts/15/16.0...
HTTPS
GET
200
applicatior
0 errors
195 requests
129.51 KB transferred
170.49 s taken (DOMContentLoaded: 1.46 s, load: 1.57 s)

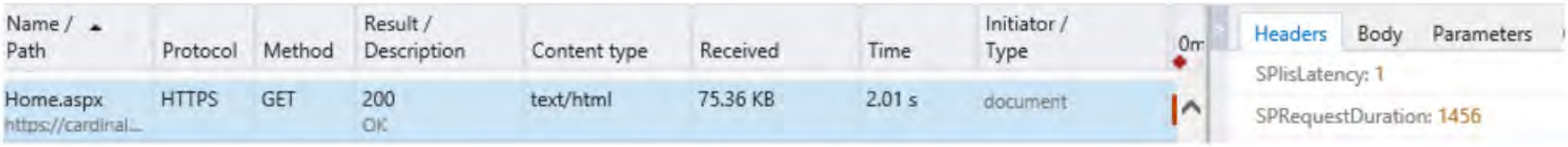
* may be due to routing issues (as in "number of hops")
* plenty of other possibilities

Demo Takeaways

HTTP Response Headers

- SPIisLatency — waiting on server - generally zero or near zero
- SPRequestDuration — time spent processing on server (in ms) - ideally low
- X-SharePointHealthScore — 0 to 10 (you want 0)

Generally speaking ...



Name / Path	Protocol	Method	Result / Description	Content type	Received	Time	Initiator / Type
Home.aspx https://cardinal...	HTTPS	GET	200 OK	text/html	75.36 KB	2.01 s	document

Headers | Body | Parameters

SPIisLatency: 1

SPRequestDuration: 1456

Time - (SPRequestDuration + SPIisLatency) = "time lost elsewhere"

* potential network latency
* may be due to routing issues (as in "number of hops")
* plenty of other possibilities

Demo Takeaways

HTTP Response

waiting on server - generally zero or

time spent processing on

So, you've concluded that your pages are slow and you have the data to prove it!

SPIisLatency is low, and
X-SharePointHealthScore is low,

but ...

So, you've concluded that your pages are slow and you have the data to prove it!

SPIisLatency is low, and
X-SharePointHealthScore is low,

but ...

SPRequestDuration is crazy high (e.g., 9000 ms)!

Repeat after me ...

"The problem probably isn't

Repeat after me ...

"The problem probably isn't SharePoint Online. It's my site."

Okay, one more time:

Repe

"The prob

SharePoir

Repeat after me ...

"The problem probably isn't SharePoint Online. It's my site."

So, who's to blame?

In all likelihood

So, who's to blame?

In all likelihood:
blame the
lousy* devs.

*Note: not all devs are lousy devs. Just the ones who create performance problems and knee-jerk into blaming Microsoft and SharePoint Online.

So, who's to blame?

In all likelihood:
blame the
lousy* devs.

*Note: not all devs are lousy devs. Just the ones who create performance problems and knee-jerk into blaming Microsoft and SharePoint Online.

nes who create performance
t and SharePoint Online.

- Compare processing and respon
  SharePoint site or page.

- In the majority of poor performance scenarios, a combination of UI/UX , client-side code additions, and questionable customization/deployment mechanisms are to blame.

- Microsoft has indicated that the slowest 1% of pages in SPO take more than 5,000ms to load - again, usually due to customizations.

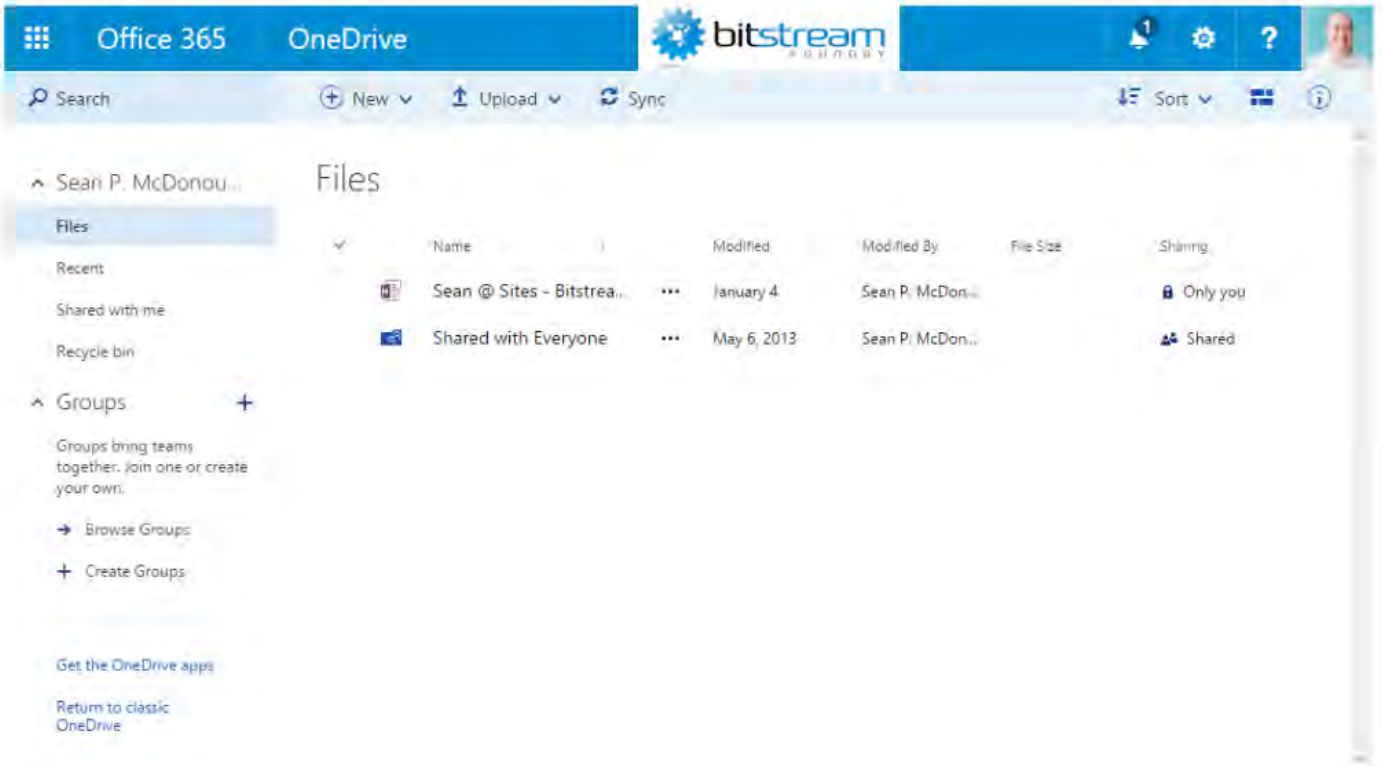
- Compare processing and response times to your problematic SharePoint site or page.

- In the majority of poor performance scenarios, a combination of UI/UX , client-side code additions, and questionable customization/deployment mechanisms are to blame.

- Microsoft has indicated that the slowest 1% of pages in SPO take more than 5,000ms to load - again, usually due to customizations.

Don't believe me?

Collect the data and validate for yourself!

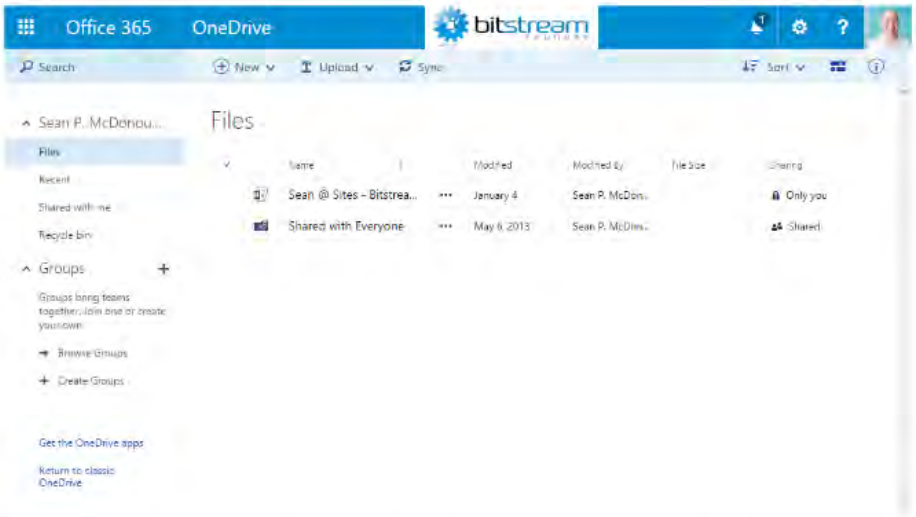


- Profile your OneDrive for Business page (it's in your MySite).
- Compare processing and response times to your problematic SharePoint site or page.

- In the majority of poor performance scenarios, a

Don't believe me?

Collect the data and validate for yourself!



DEMO

- Profile your OneDrive for Business page (it's in your MySite).
- Compare processing and response times to your problematic SharePoint site or page.

- In the majority of poor performance scenarios, a combination of UI/UX , client-side

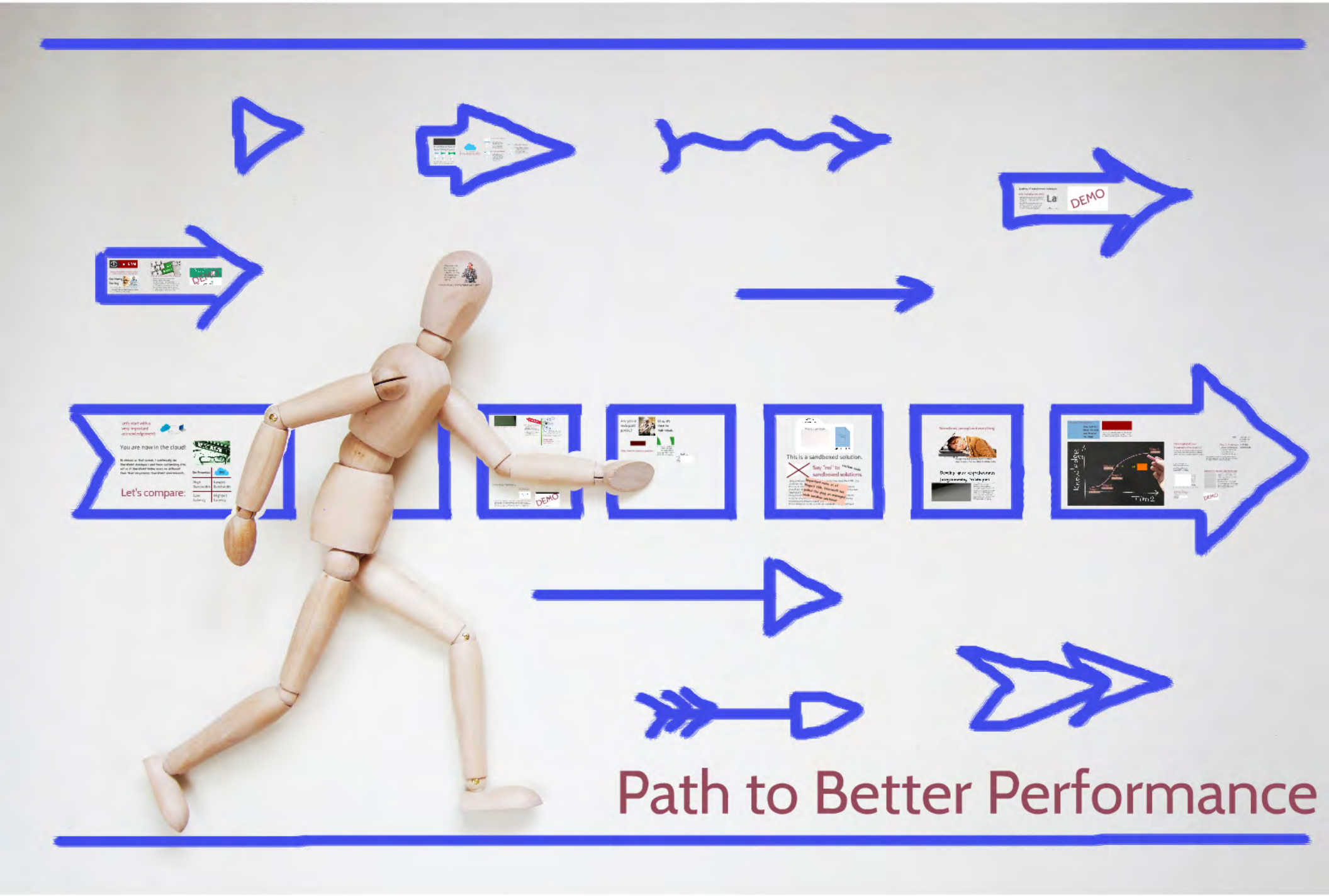
Don't believe me?

"Okay, yeah - my OneDrive for Business page is really fast ... but my SharePoint pages are completely choking."

u're probably thinking "What can I do

"Okay, yeah - my OneDrive for Business page is really fast ... but my SharePoint pages are completely choking."

You're probably thinking **"What can I do?"**


DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

Let's start with a very important acknowledgement:

Let's start with a very important acknowledgement:

You are now in the cloud!

As obvious as that sounds, I continually see

acknowledgement:

You are now in the cloud!

As obvious as that sounds, I continually see SharePoint developers and those customizing sites act as if SharePoint Online were no different than their on-premises SharePoint environments.

very important
acknowledgement:

You are now in the cloud!

As obvious as that sounds, I continually see SharePoint developers and those customizing sites act as if SharePoint Online were no different than their on-premises SharePoint environments.

Let's compare:

On-Premises	SPO
High Bandwidth	Low(er) Bandwidth
Low Latency	High(er) Latency

Failing to acknowledge the "we're in the cloud now" reality leads to a problem I simply call ...

Too Many

Failing to acknowledge the "we're in the cloud now" reality leads to a problem I simply call ...

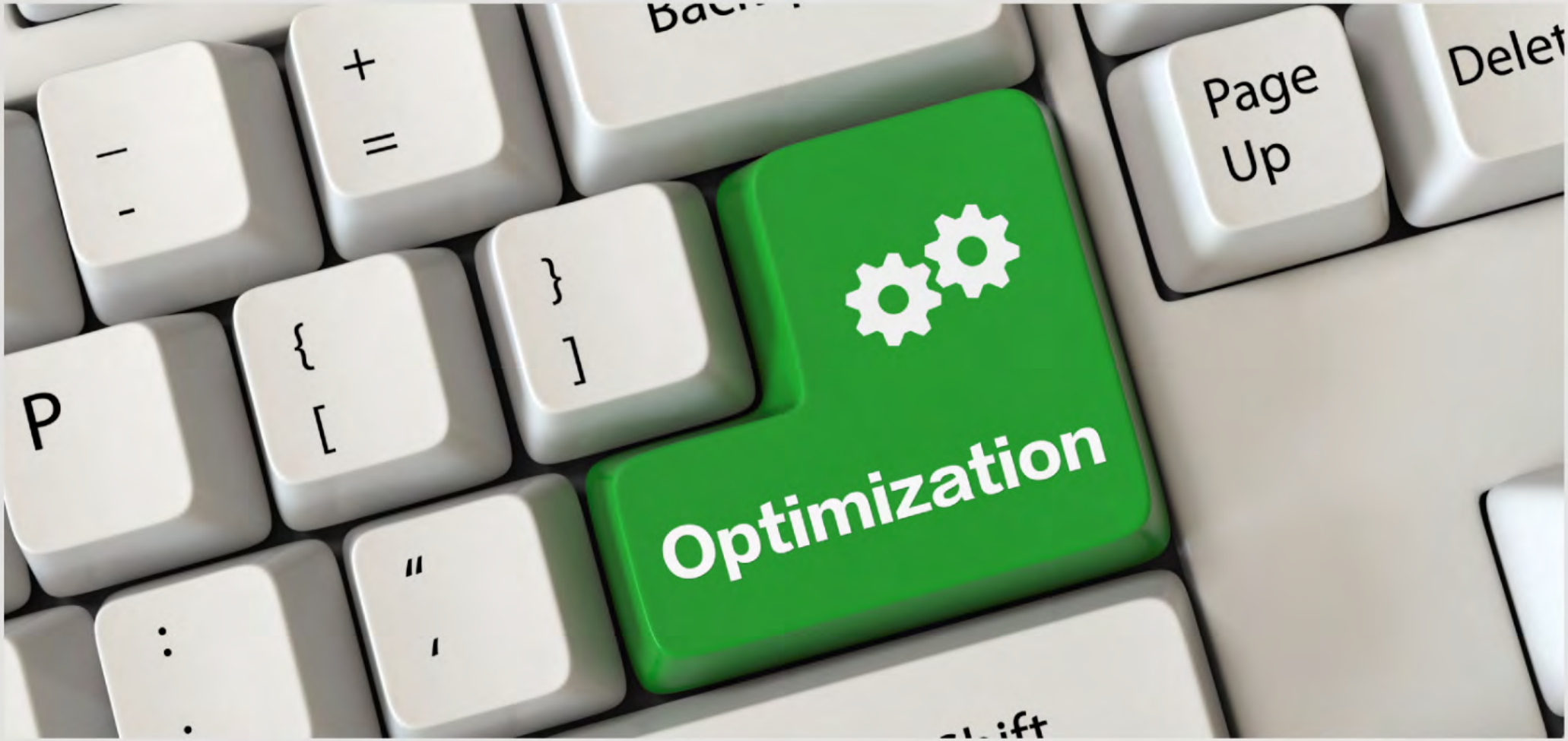
Too Many, Too Big

- Too many calls are made to the server.

now" reality leads to a problem I simply call ...

Too Many, Too Big

- Too many calls are made to the server.
- Too many files are referenced on pages.
- The files in-use are too large.

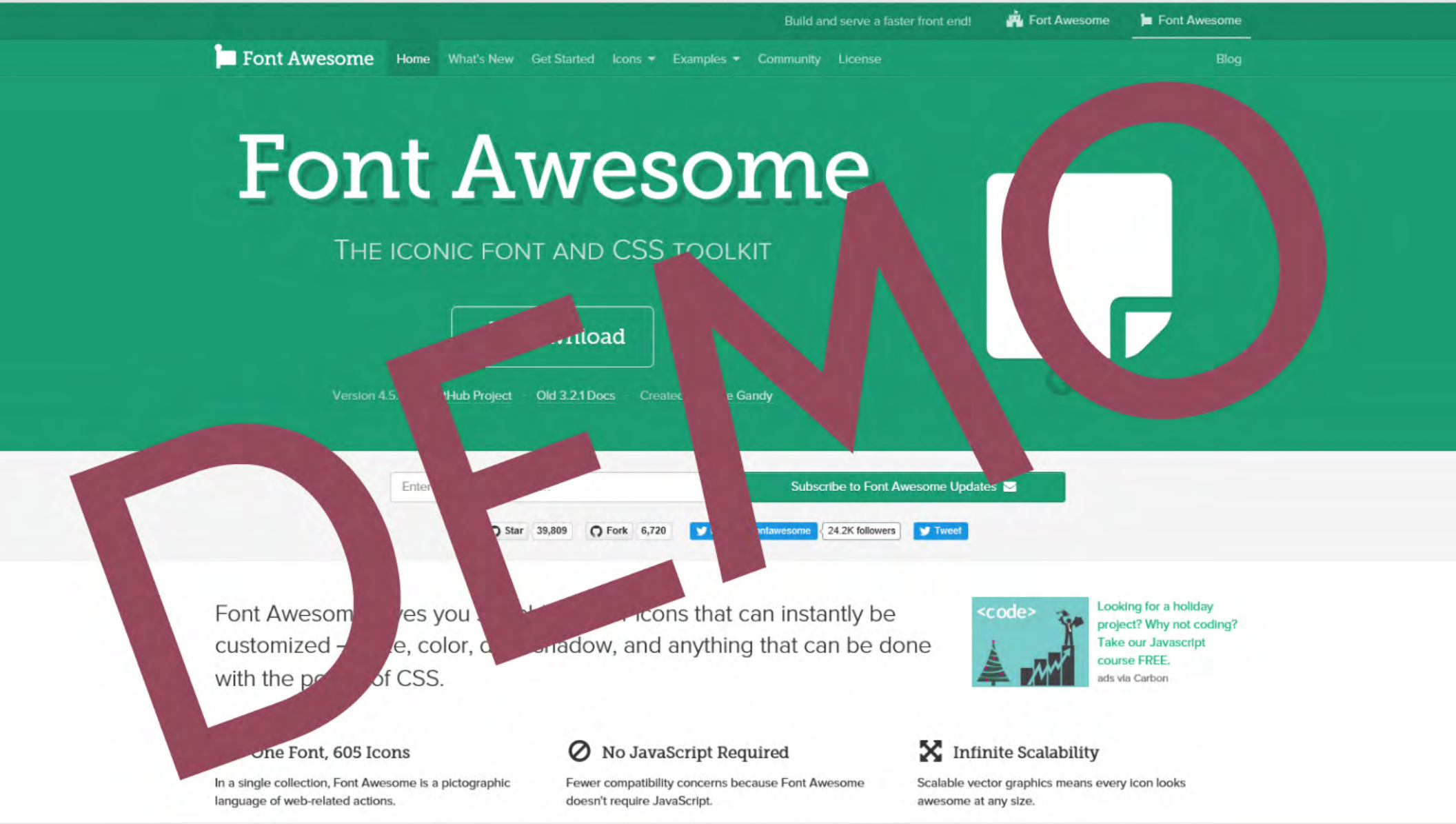


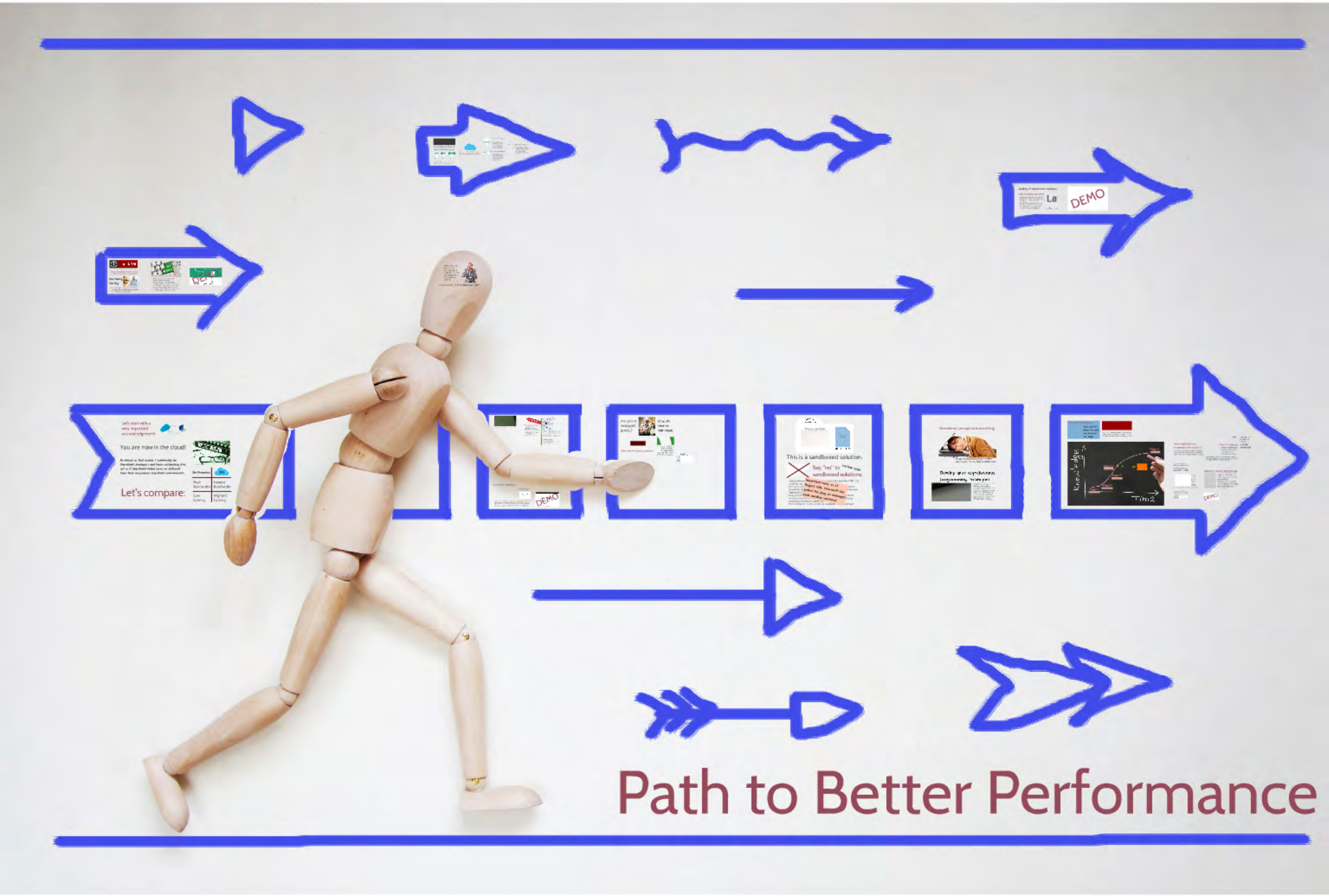
Consider one or more of the following:

- Minify files, especially JavaScript files.
- Resize images to usage sizes

Consider one or more of the following:

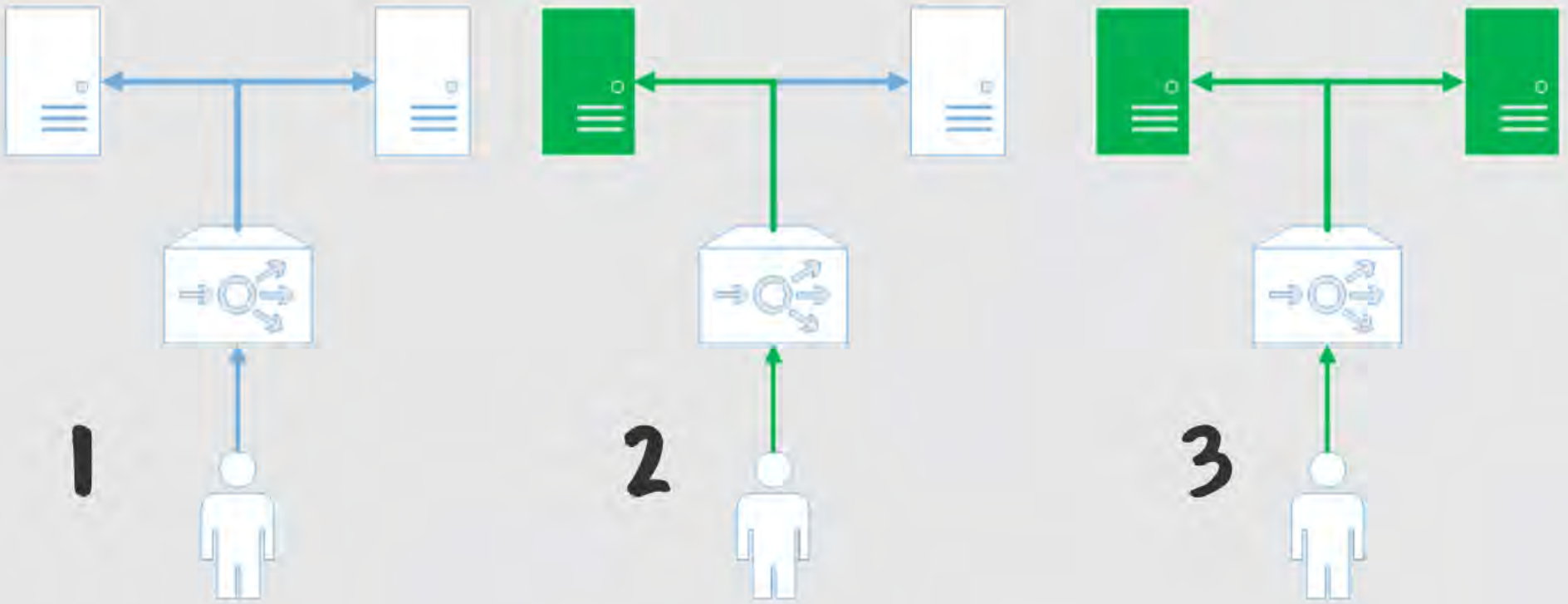
- Minify files, especially JavaScript files.
- Resize images to usage sizes.
- Compress images (more) aggressively.
- Use sprite sheets to reduce the actual number of HTTP requests needed to retrieve images.
- Use SharePoint's Image Rendition service.
- Leverage a toolkit like Font Awesome in place of individual icons and associated files.


Build and serve a faster front end!
Font Awesome
Font Awesome
Font Awesome
Home
What's New
Get Started
Icons
Examples
Community
License
Blog
Font Awesome
THE ICONIC FONT AND CSS TOOLKIT
Subscribe to Font Awesome Updates
Star 39,809
Fork 6,720
24.2K followers
Tweet
DEMO
<code>
Looking for a holiday project? Why not coding? Take our Javascript course FREE.
ads via Carbon
One Font, 605 Icons
In a single collection, Font Awesome is a pictographic language of web-related actions.
No JavaScript Required
Fewer compatibility concerns because Font Awesome doesn't require JavaScript.
Infinite Scalability
Scalable vector graphics means every icon looks awesome at any size.

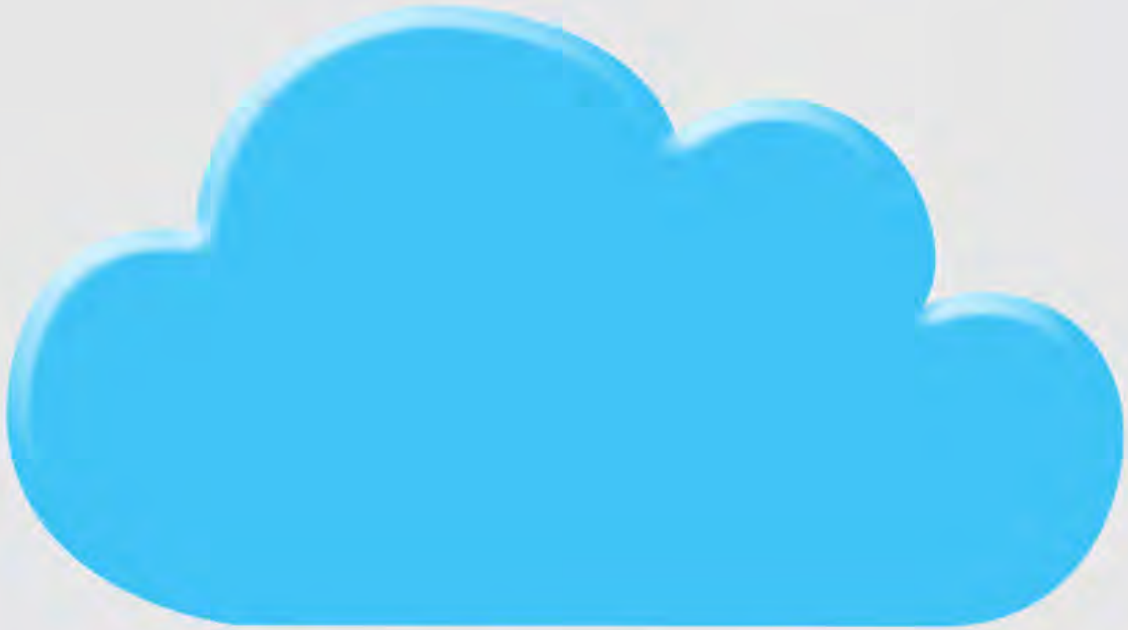

DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

Conventional wisdom says caching is good.

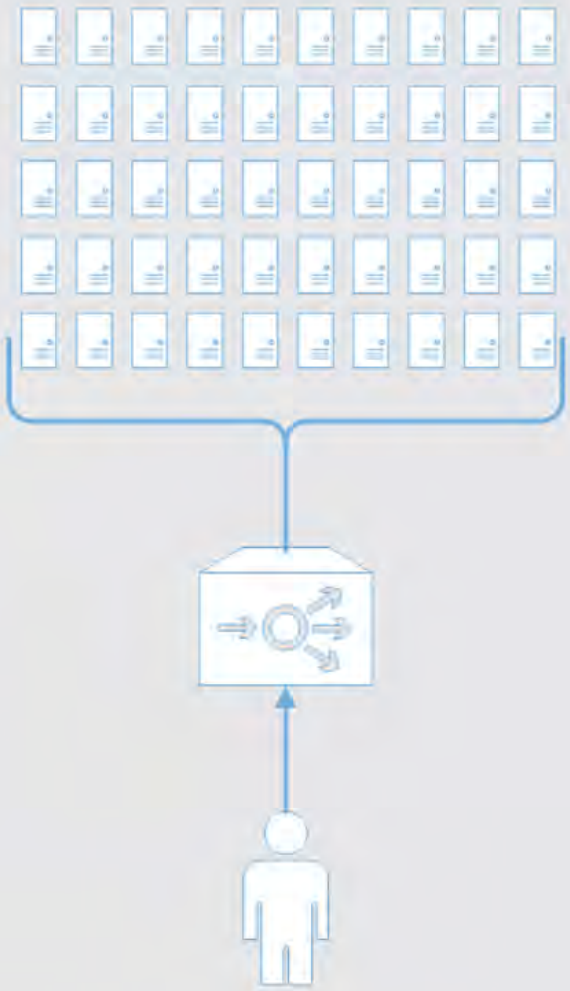
says caching is good.



After just a few requests, the on-premises Object Cache can be "ready for action."



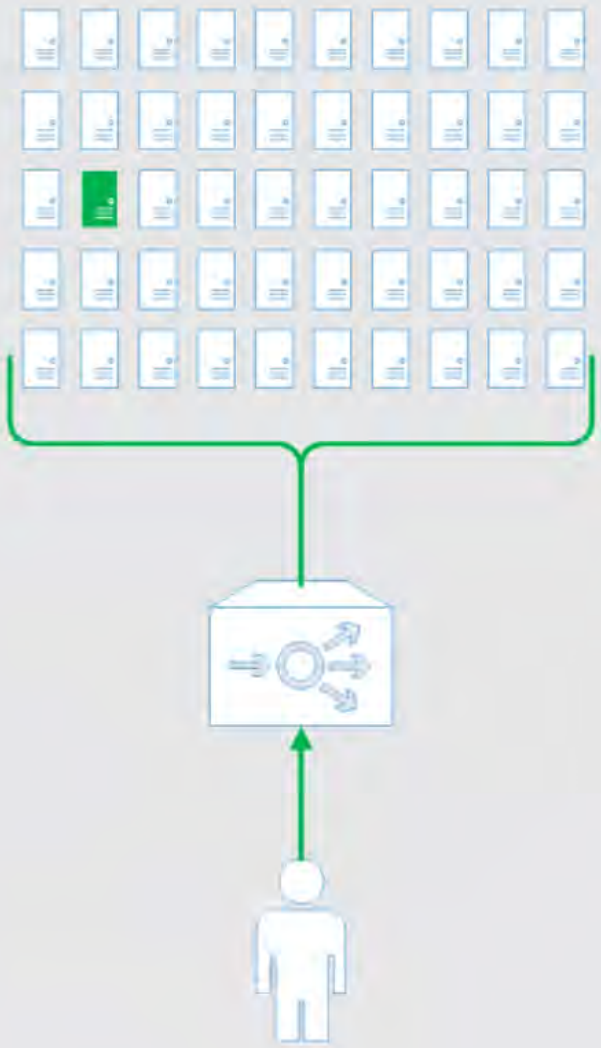
In the cloud, the caching equation (for per-server memory-based caches like the Object Cache) works out a bit differently.



User's Initial Request

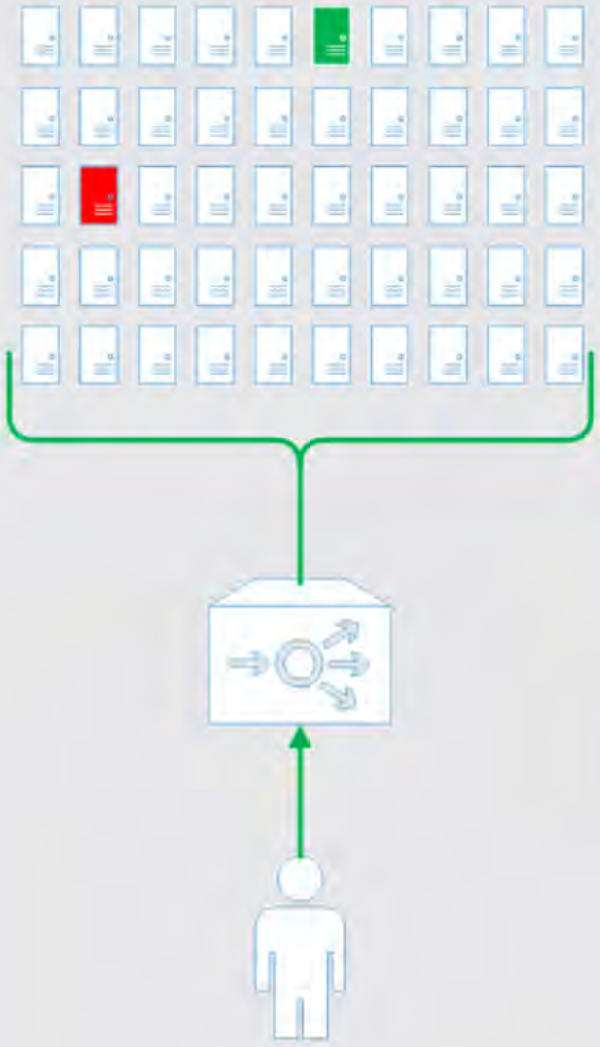
- First thing to note: the number of WFEs tends to be *much* higher in the cloud versus on-premises.

User's Second Request



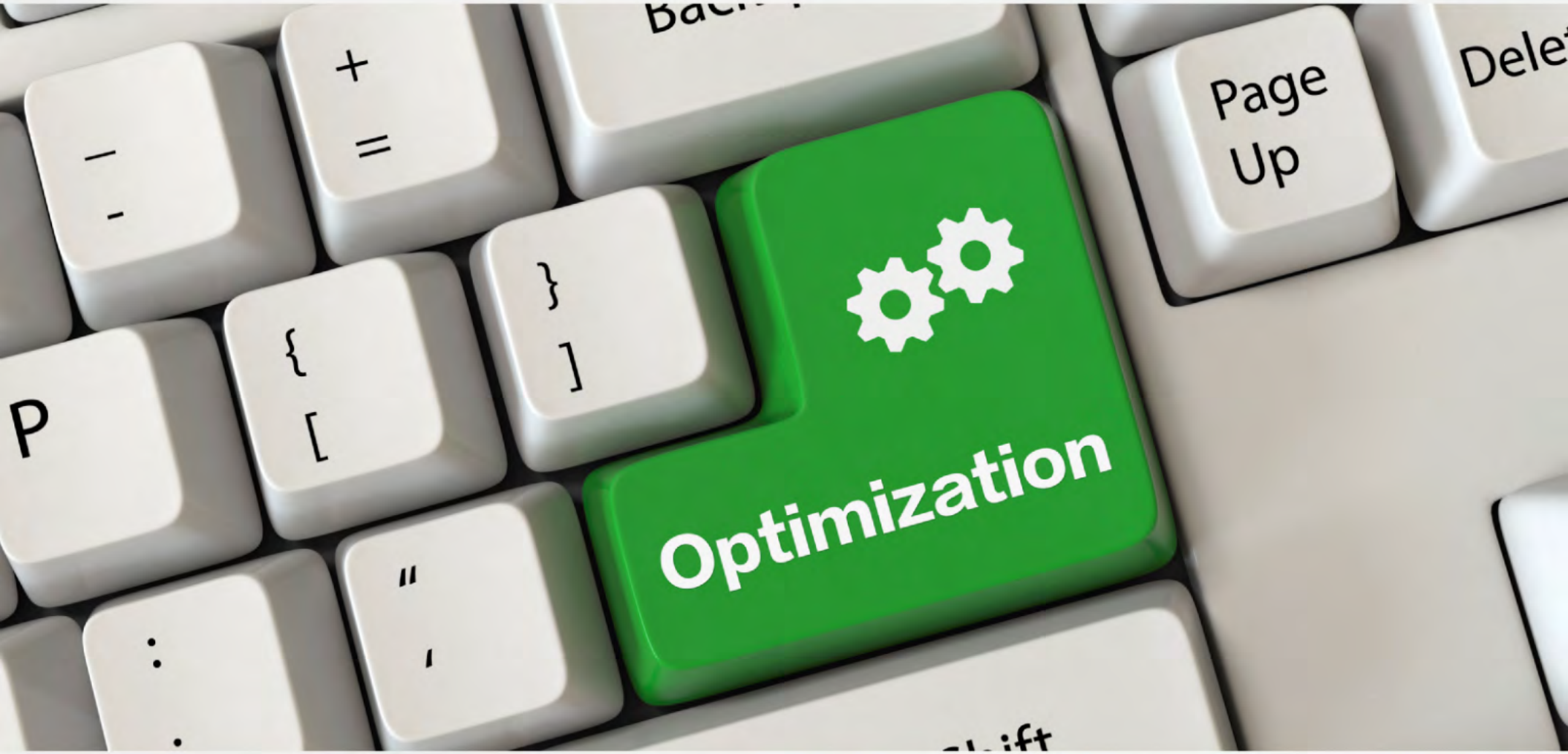
User's Second Request

- No affinity is in use, so the chance of a user hitting the same server again is dramatically less than the on-premises scenario.



Subsequent Requests

- Same reduced chance of hitting the WFE last visited
- Memory pressure causes much more frequent cache ejections versus on-premises.



Two significant adjustments can be made.

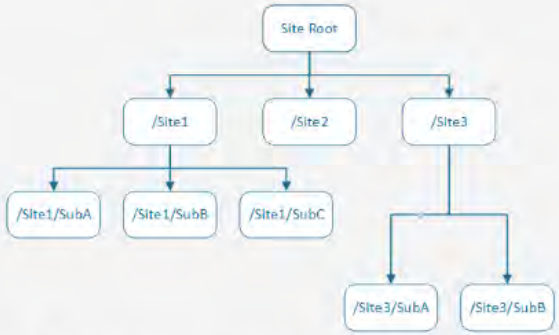
* These sitemaps are then stored in the Object

Navigation style has a huge impact on performance.

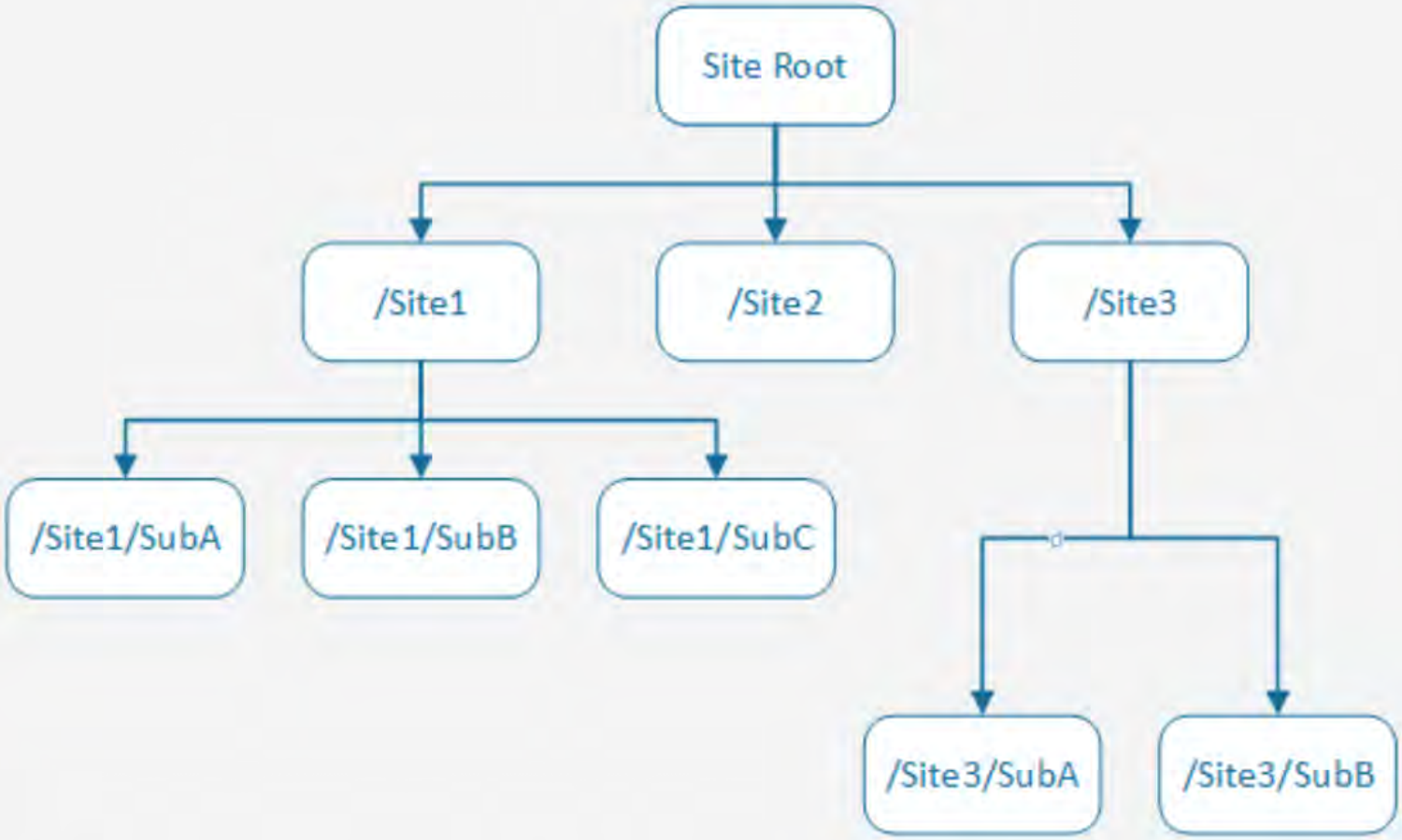
Using structural navigation is the default (but poor) choice for complex site hierarchies in the cloud.

* building each site node generates roughly 8 SQL Server round trips
* These sitemaps are then stored in the Object Cache on WFEs

Navigation style has a huge impact on performance.



8 site nodes/~64 SQL calls



8 site nodes/~64 SQL calls

Better Options for Navigation

- Managed Navigation (i.e., using a term set to drive navigational structures) can significantly improve page performance.
  note: the SharePoint Server Publishing Infrastructure site collection Feature must be enabled to use Managed Navigation
- Search-driven navigation leverages SharePoint's Search index and the process of client-side navigational rendering to dramatically speed things up.
  note: implementation is non-trivial and less customizable

Using structural navigation is the default (but poor) choice for complex site hierarchies in the cloud.

* building each site node generates roughly 8 SQL Server round trips
* These sitemaps are then stored in the Object Cache on WFEs

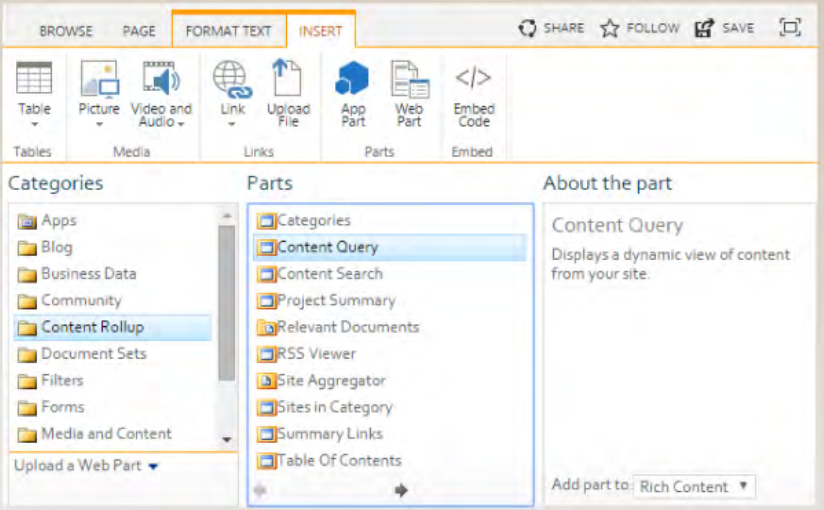
Navigation style

As was pointed-out in the navigational scenario,
Search can be used to boost performance significantly.

In the cloud, the CQWP can cause some signif

ut in the navigational scenario,
ed to boost performance significantly.

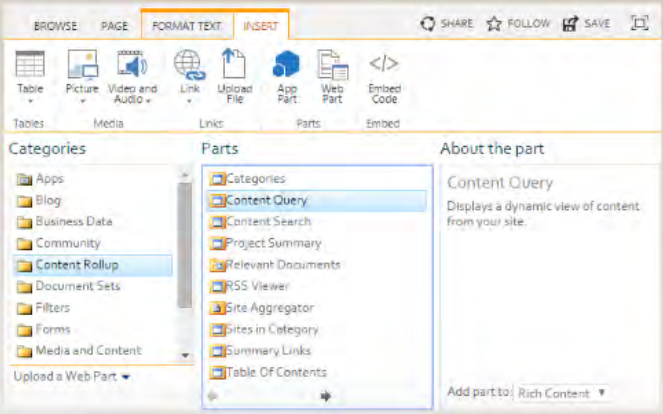
Do you like the Content Query Web Part (CQWP)?



the CQWP can cause some significant performance hits.

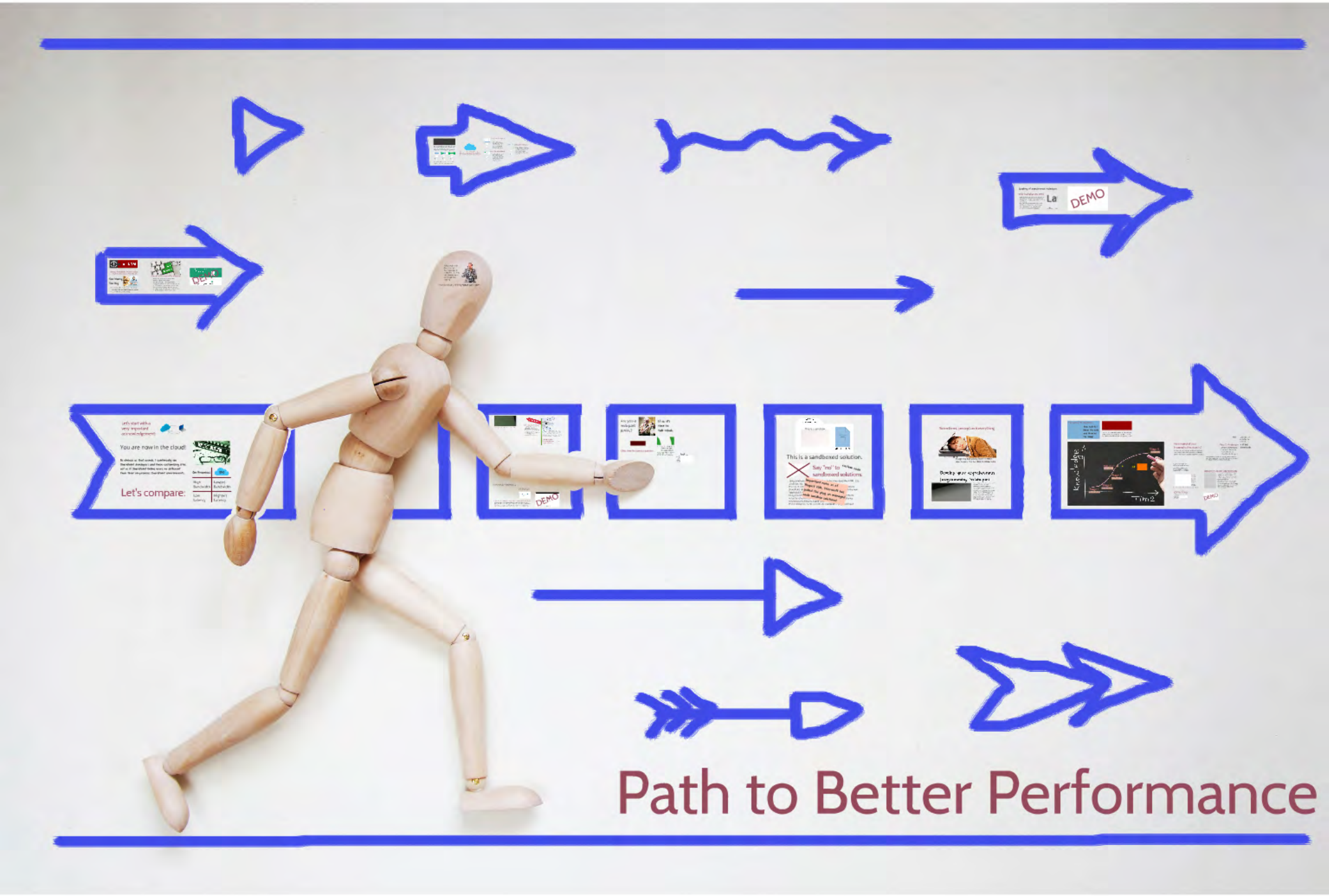
As was pointed-out in the navigational scenario,
Search can be used to boost performance significantly.

Do you like the Content Query Web Part (CQWP)?



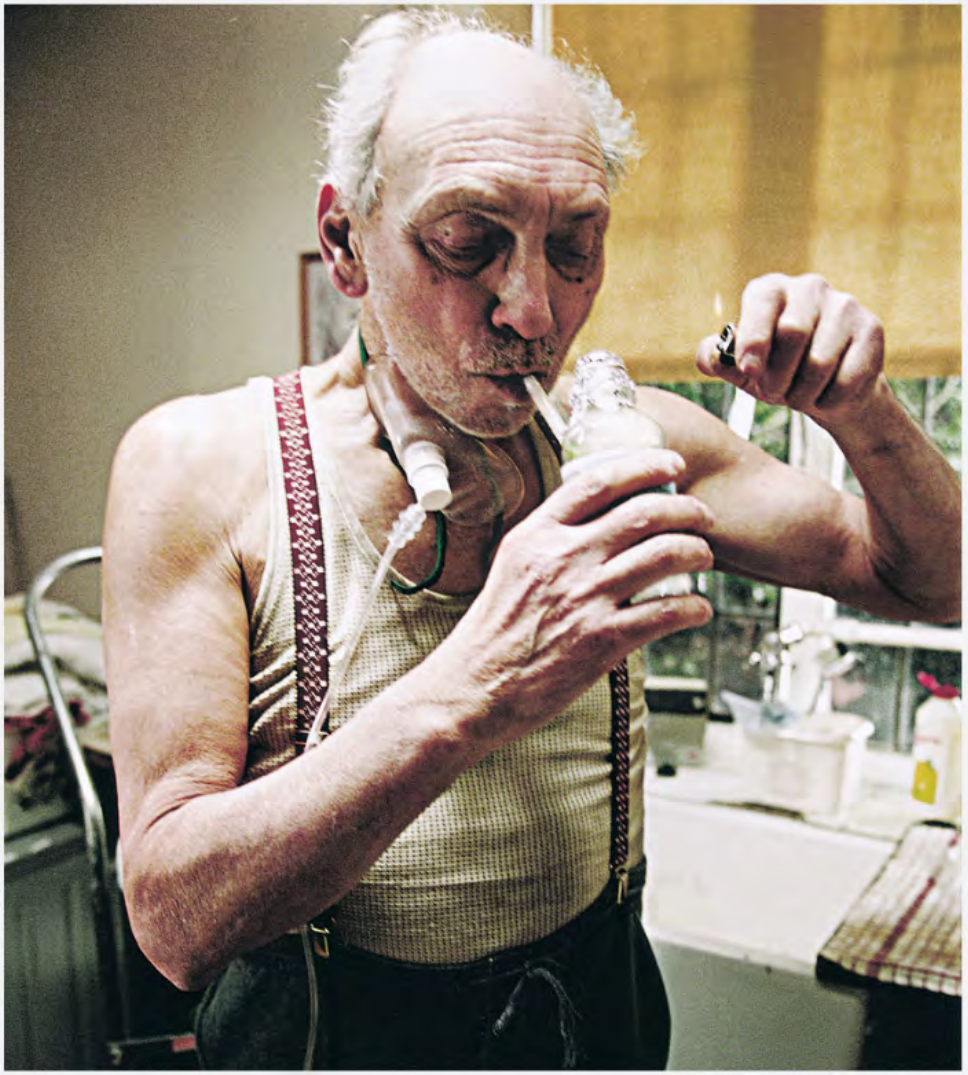
In the cloud, the CQWP can cause some significant performance hits.

- The CQWP performs expensive cross-list and cross-site queries at run-time.
- The CQWP relies on the Object Cache to store results for acceptable performance.
- The Content Search Web Part (CSWP) provides options that are similar to the CQWP (and in a number of ways, more powerful) and uses Search so it's FAST!

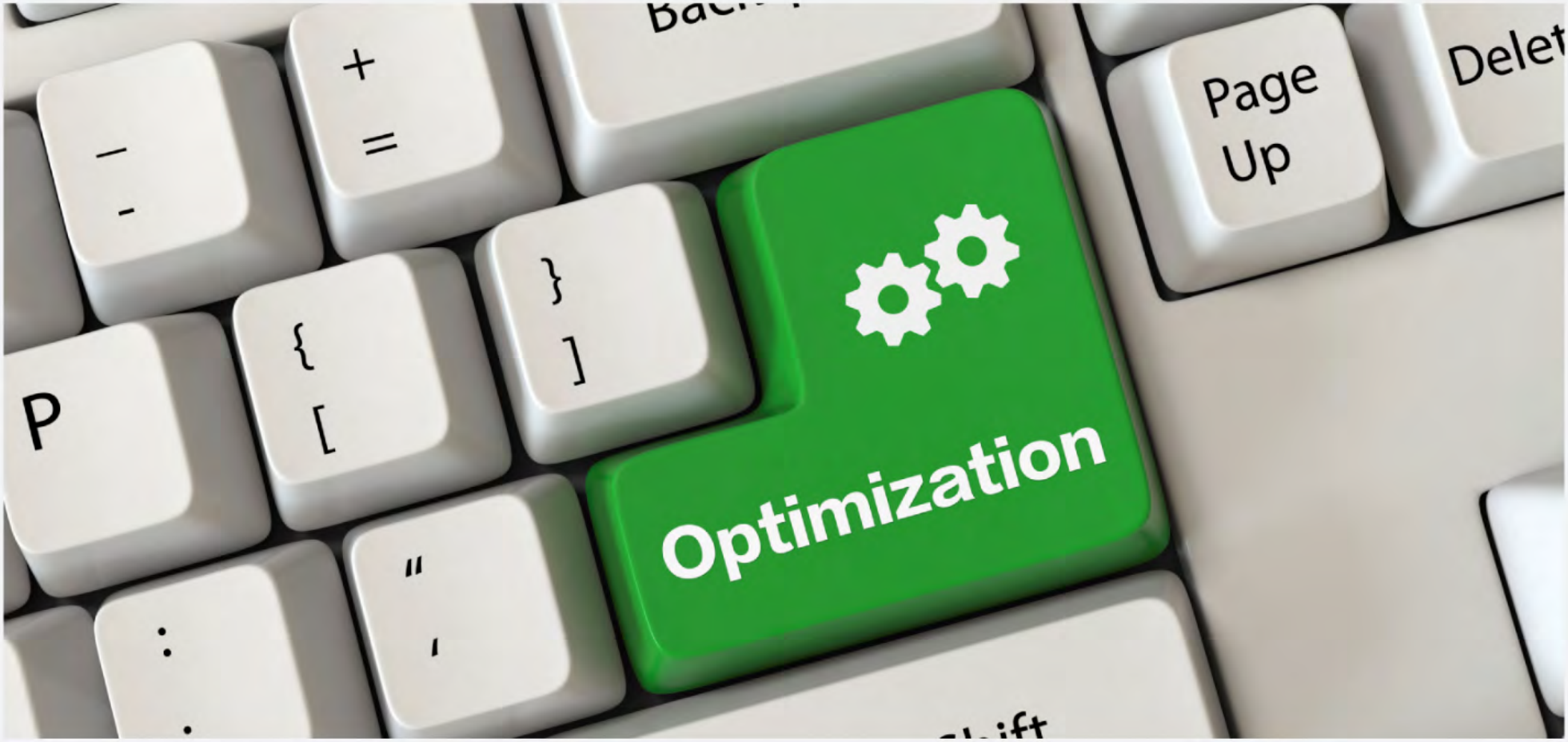

DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

Okay, time for a serious question ...

Are you a web part junkie?

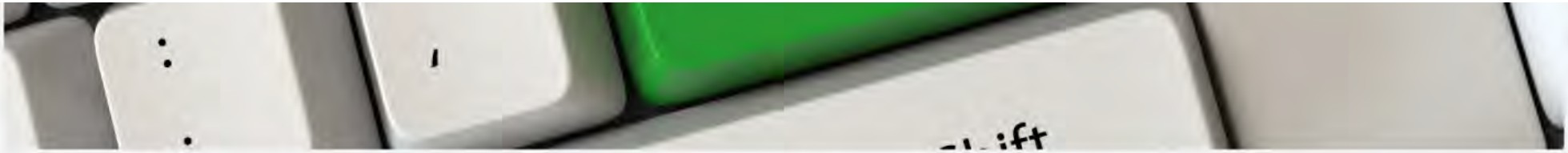


If so, it's time to talk rehab.



What's the alternative to freebasing web parts?

There's no single



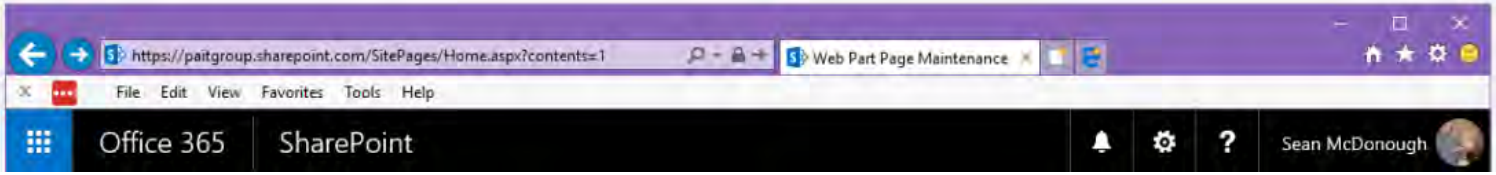
What's the alternative to freebasing web parts?

There's no single (or simple) answer.

Generally speaking, consider leveraging client-side code (JavaScript) and asynchronous techniques - both of which we'll discuss soon.

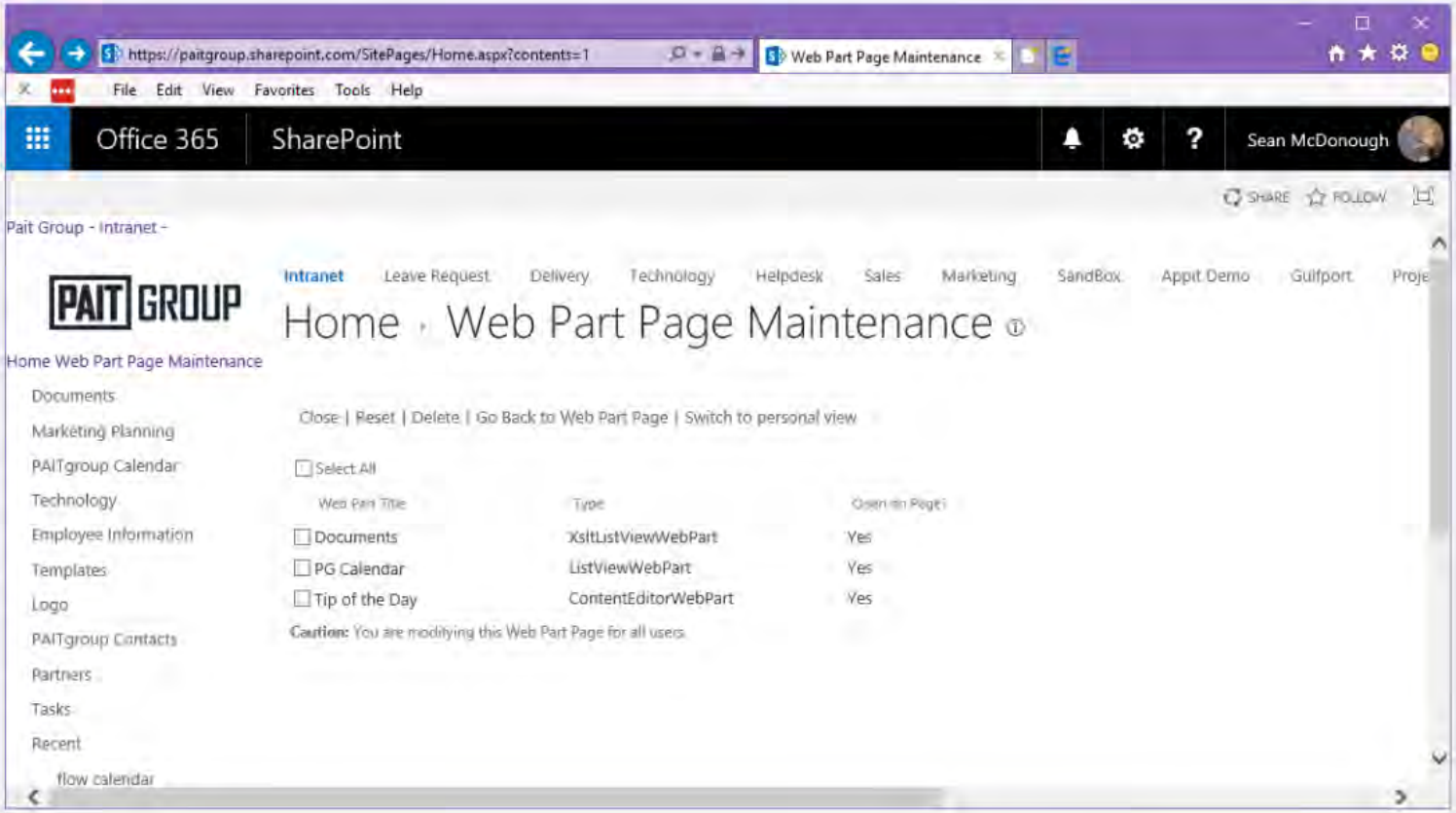
but ...

Be sure to put problem pages



but ...

Be sure to put problem pages in web part maintenance view with **?contents=1** to find web parts which are closed but not deleted!



https://paitgroup.sharepoint.com/SitePages/Home.aspx?contents=1

File Edit View Favorites Tools Help

Office 365 | SharePoint

Intranet -

Intranet Leave Request Delivery

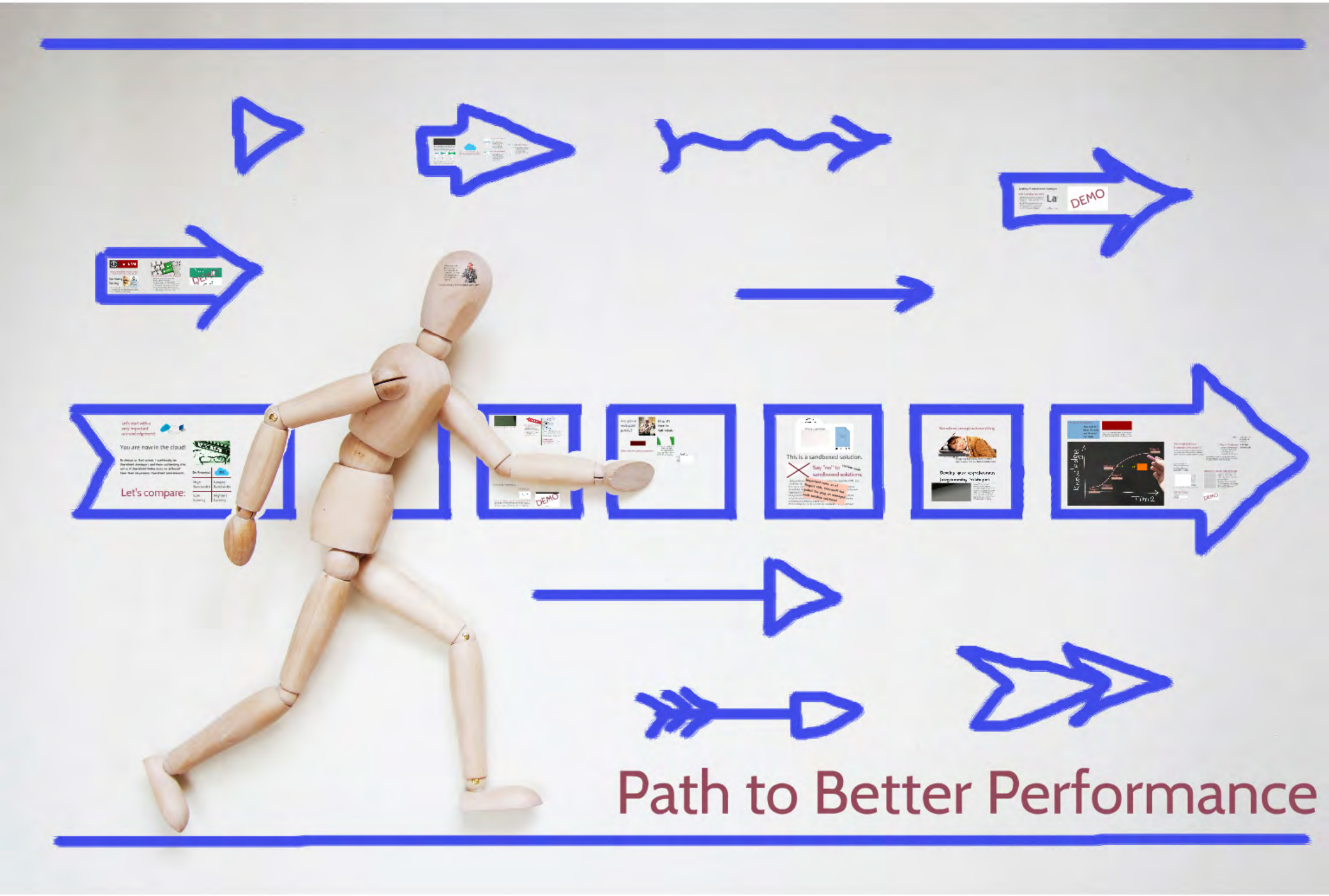
Home ▸ Web Part Page Maintenance

Close | Reset | Delete | Go Back to Web Part Page | Switch to personal view

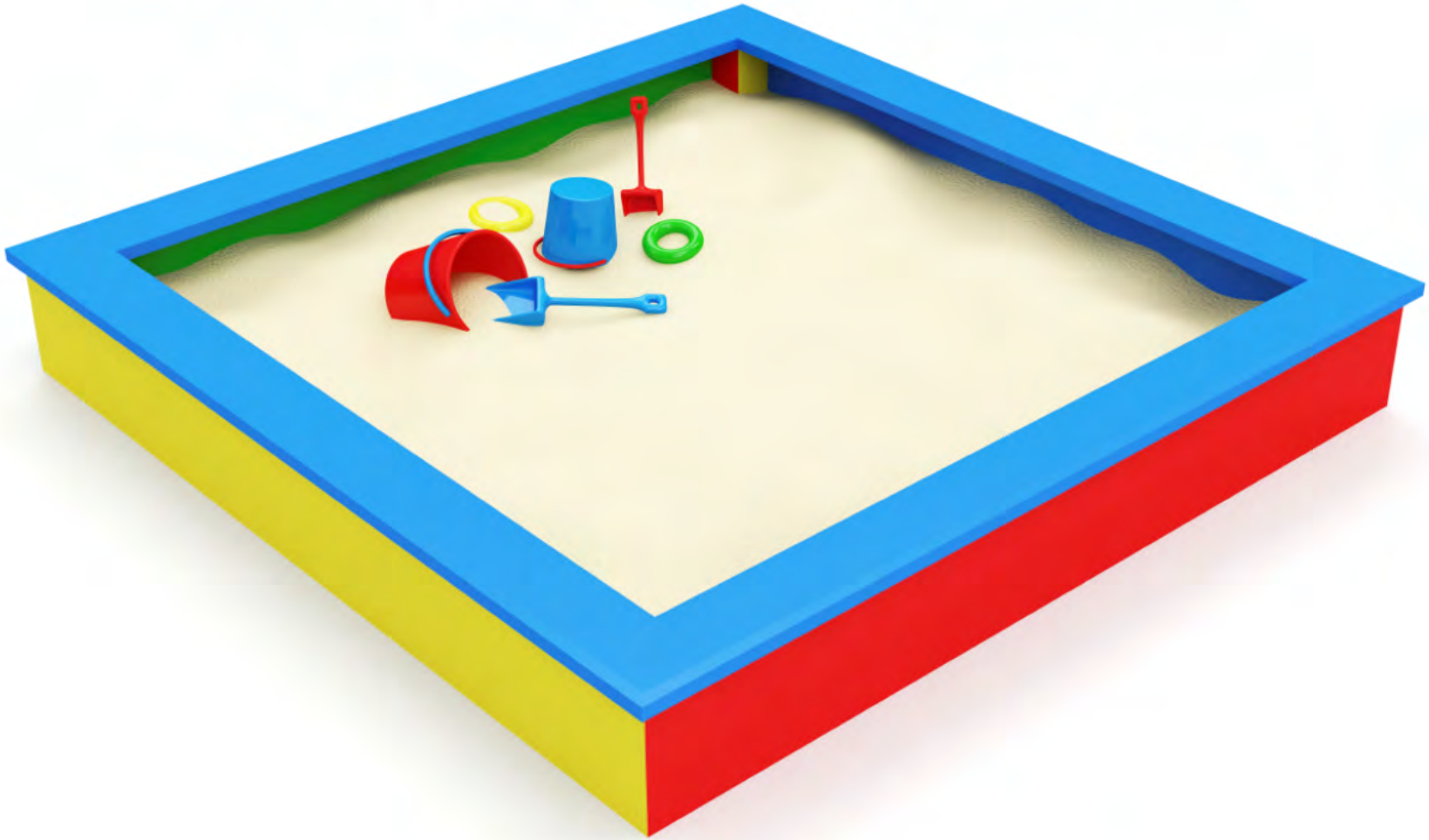
Select All

Web Part Title	Type	Open on Page?
Documents	XsltListViewWebPart	Yes
PG Calendar	ListViewWebPart	Yes
Tip of the Day	ContentEditorWebPart	Yes

Caution: You are modifying this Web Part Page for all users.

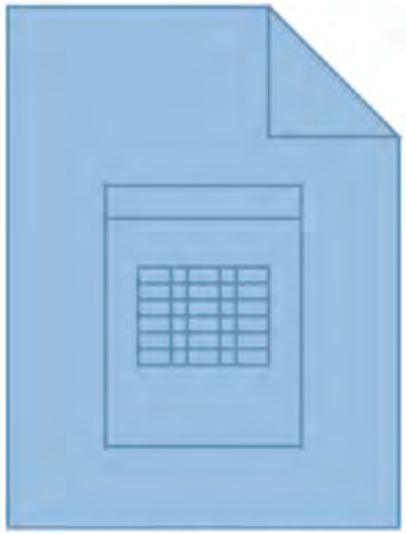

DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

This is a sandbox.



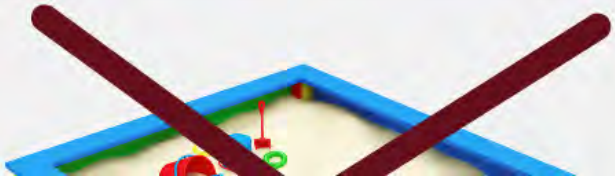
WSP File
(CAB structure)

This is a sandbox.



MANIFEST.XML

This is a sandboxed solution.



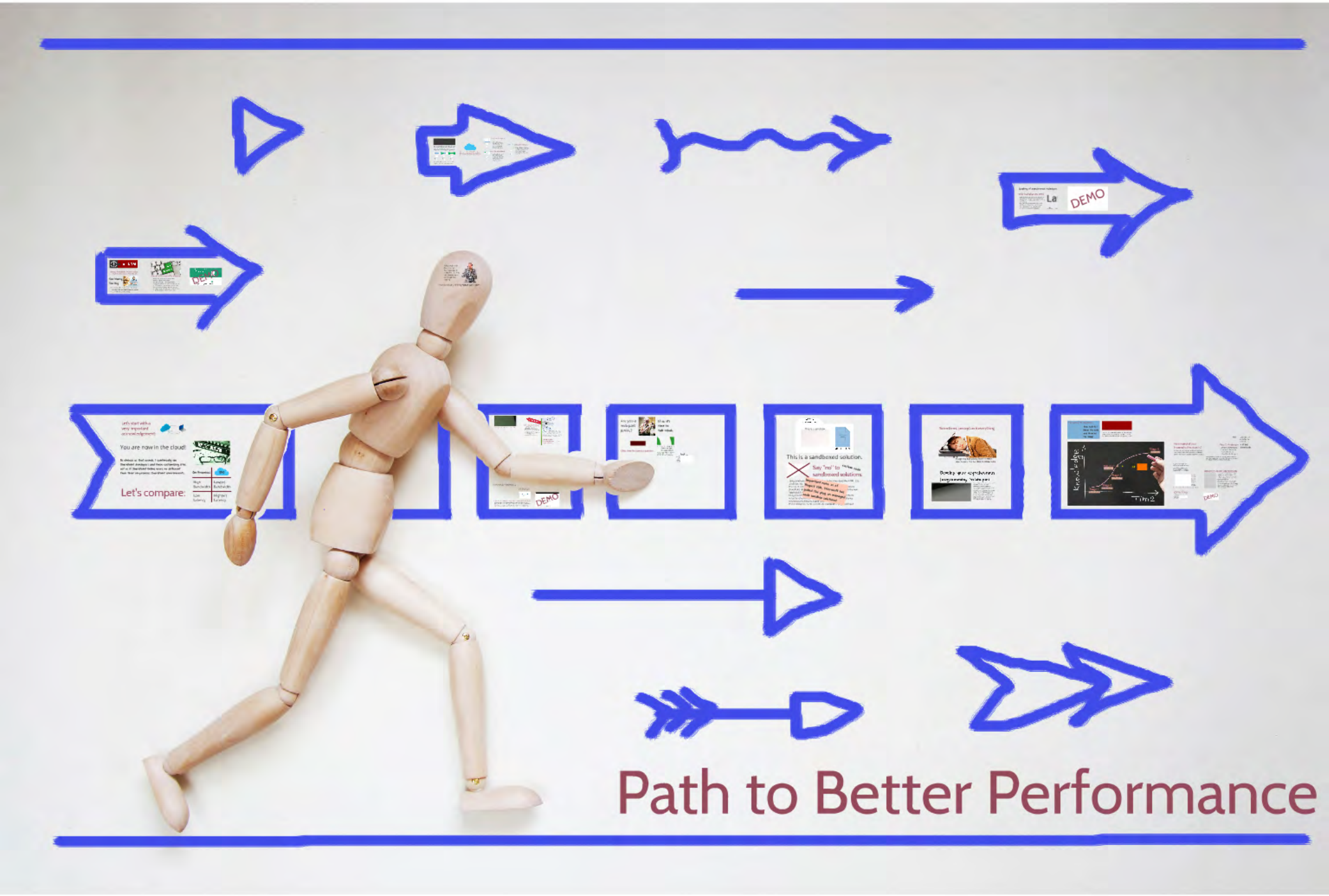
Say "no" to custom

This is a sandboxed solution.

Say "no" to custom code sandboxed solutions.

- Using sandboxed [illegible] declarative assets like HTML, CSS, JavaScript, etc., i[illegible]
- The use of custo[illegible]ated in SharePoint 201[illegible]ly the new "App Model").*
- A big problem [illegible]rtificate validation checks occur [illegible]dboxed solutions containing managed code.
- These validation checks can introduce ***seconds*** of pag[illegible]verhead!

Important note: as of August 2016, Microsoft has pulled the plug on managed code sandbox solutions!


DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

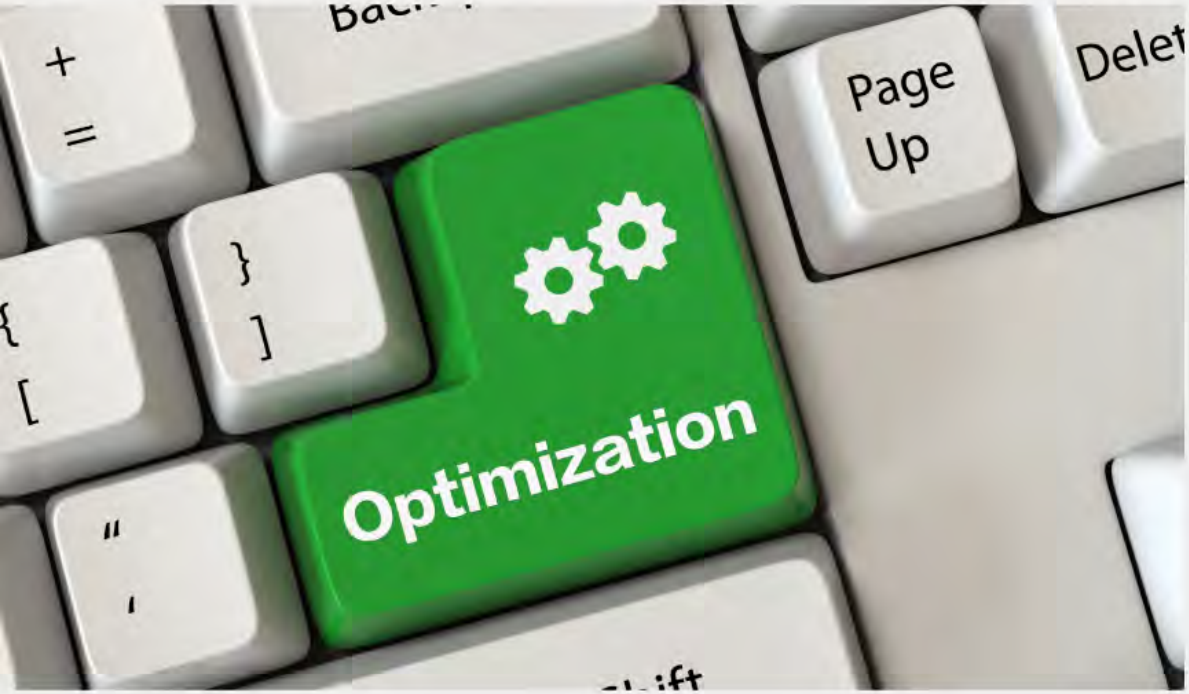
Sometimes, perception is everything.

A page may load quickly, but if it FEELS slow to users, it is the SAME AS BEING SLOW.

Sometimes, perception is everything.

A page may load quickly, but if it FEELS slow to users, it is the SAME AS BEING SLOW.

Develop your asynchronous programming techniques

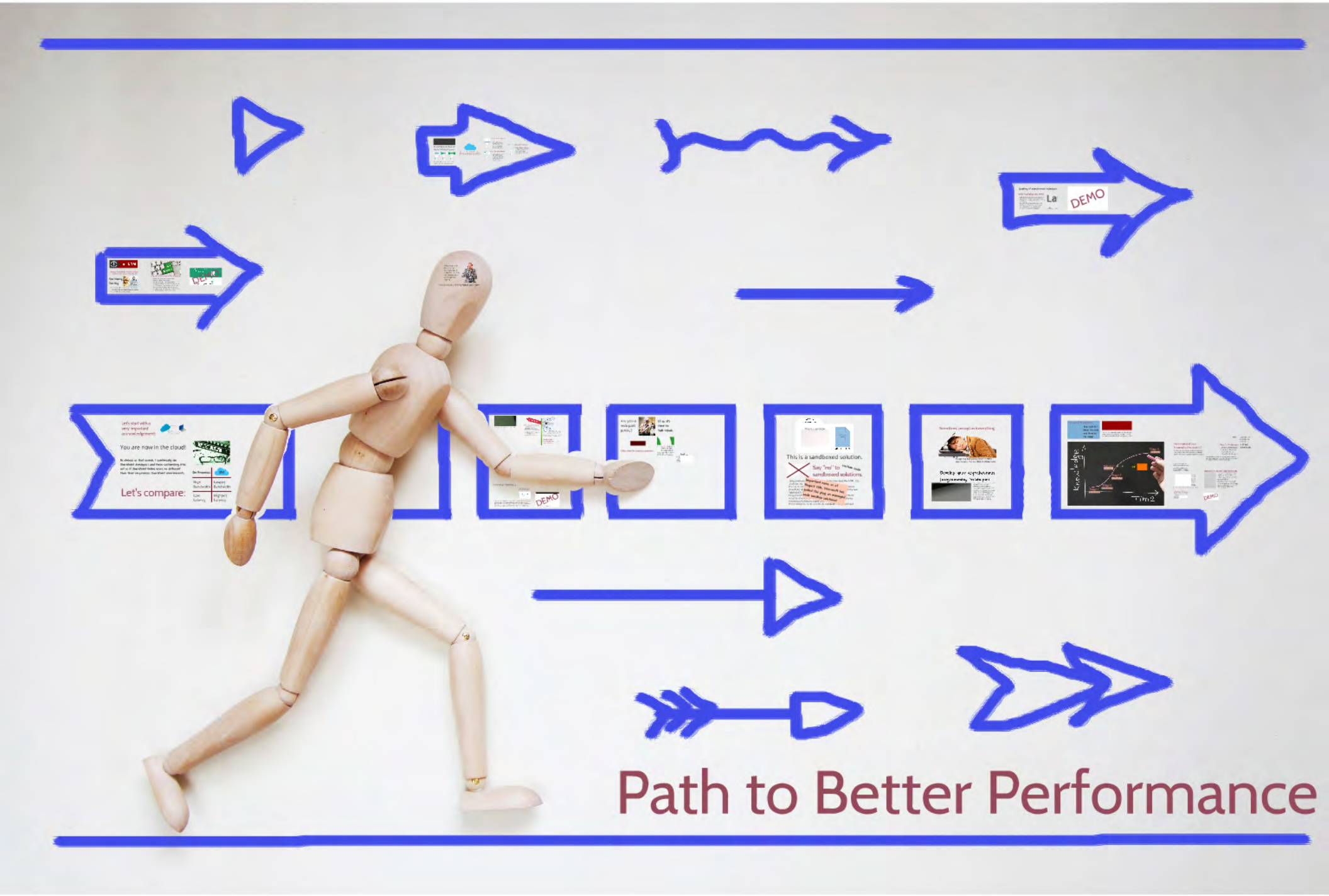


- You can't block a browser's main thread of execution, so leveraging async development patterns is essential.
- Async programming is made much easier in jQuery using promises. Promises approximate a synchronous programming model under asynchronous conditions.
- Certain web parts (e.g., the CSWP) also allow you to set their (a)sync behavior.
- Good use of async techniques make pages ***appear*** to load faster ... and as we discussed, perception is everything.

Speaking of asynchronous techniques:

Only load what you need.

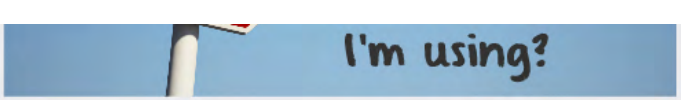
- Instead of fetching everything at once within the context of the initial page load, retrieve the page with only the payload that's needed immediately.
- (Lazy) load images and other items "below the fold" only if users start scrolling down and will see them (e.g., Facebook and LinkedIn's "forever-scrolling" pages).


DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

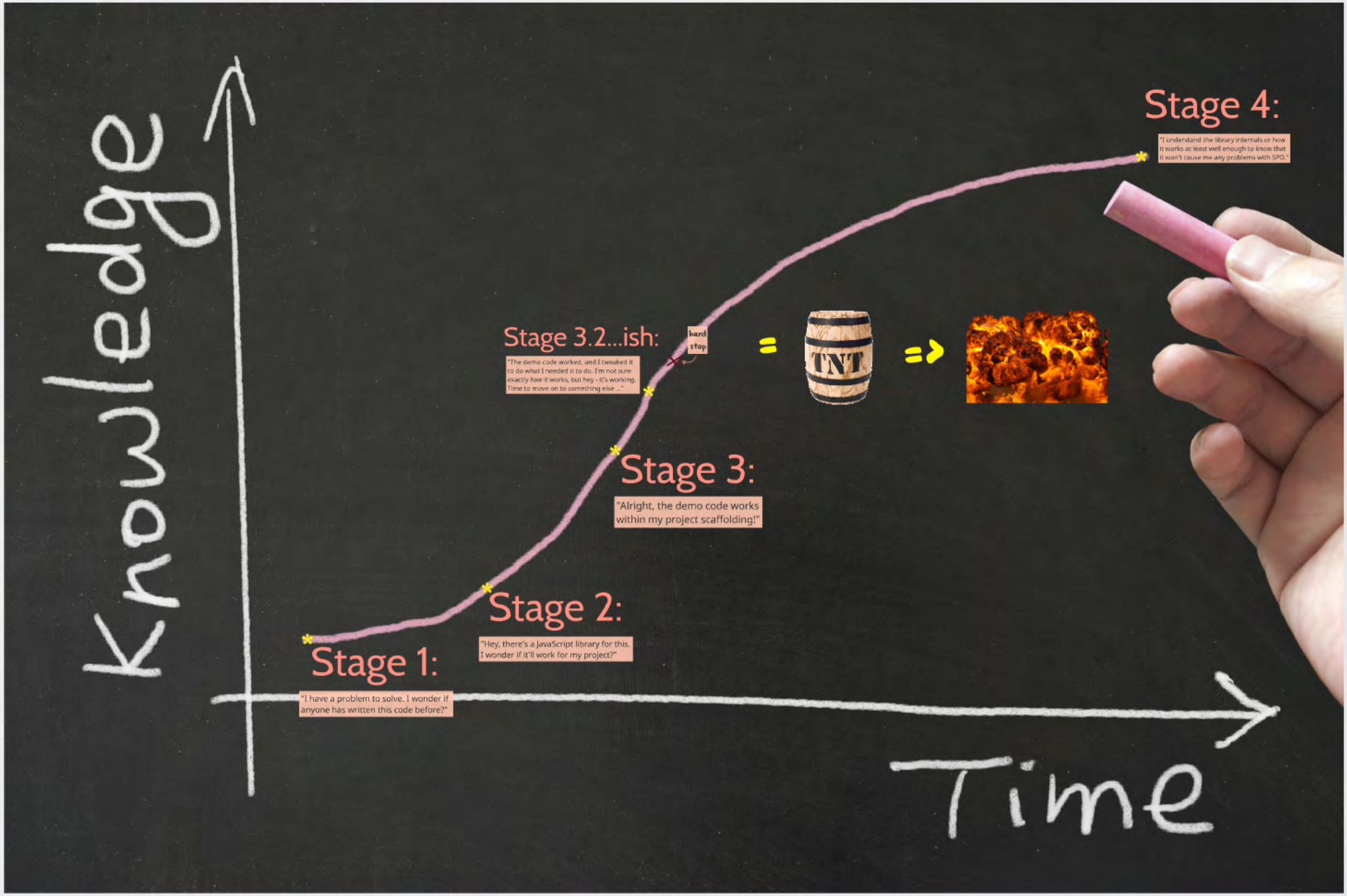
Ask yourself this question:

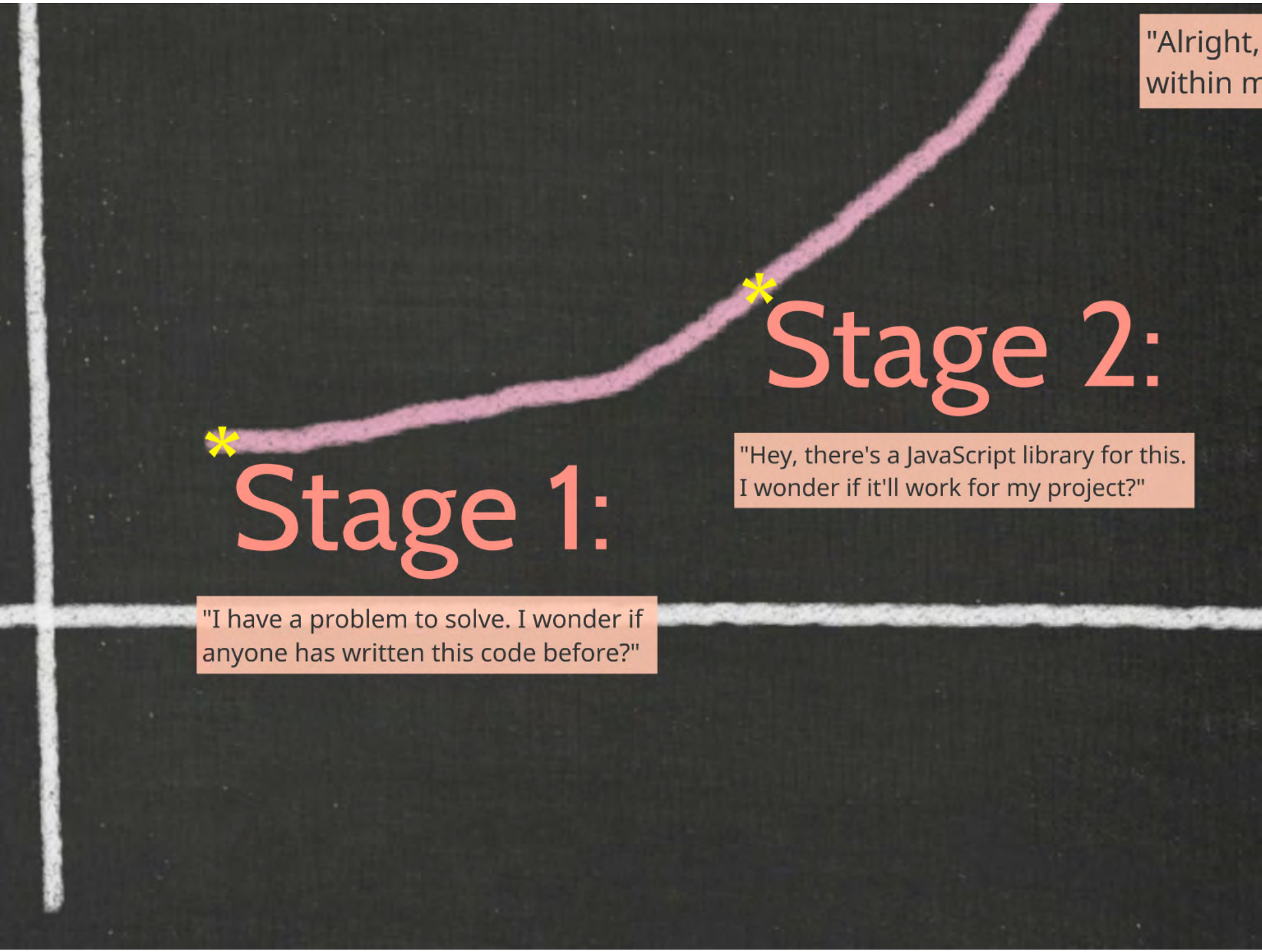
How well do I know the code and libraries I'm using?

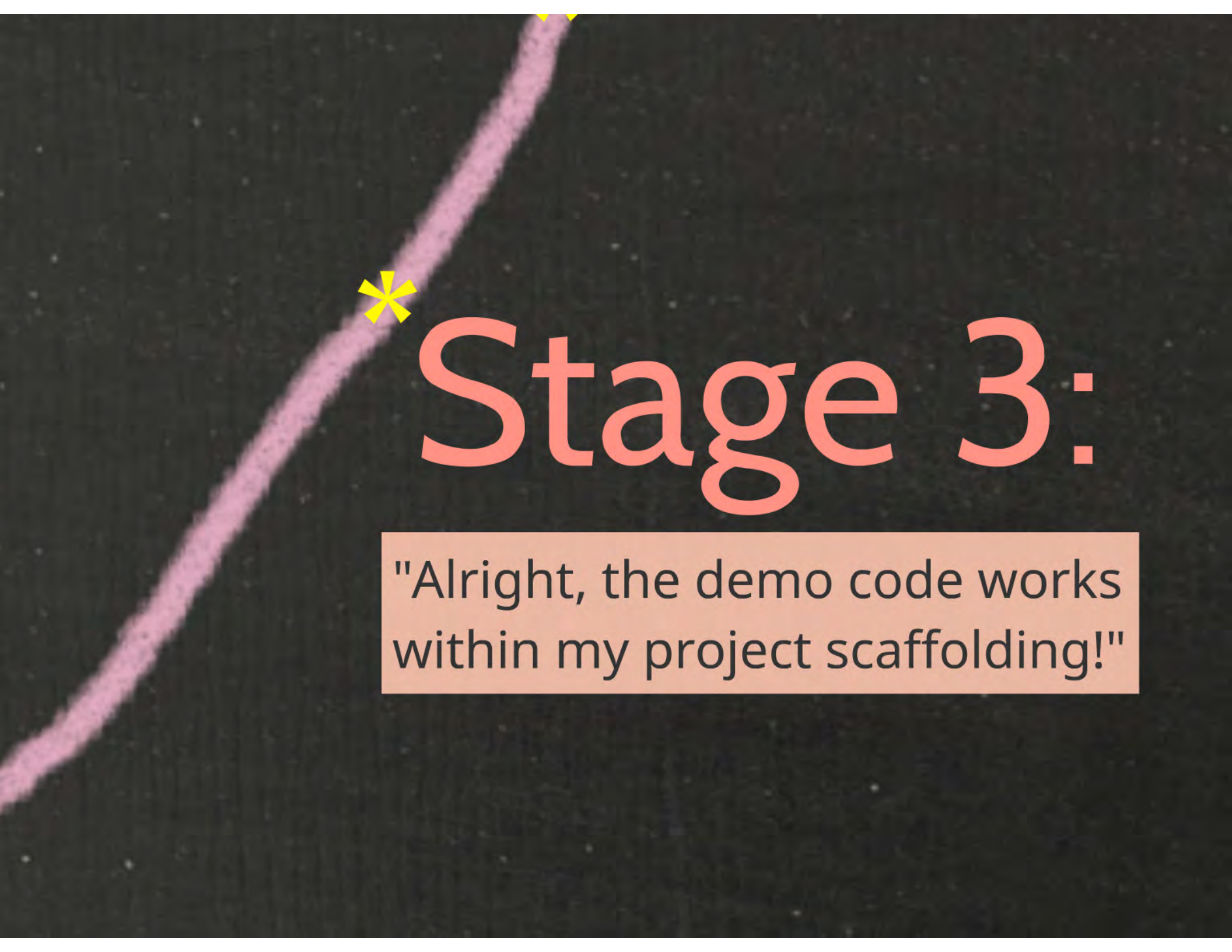
Reason I ask: when troubleshooting performance issues, I commonly encounter a pattern that can be illustrated with the following diagram of stages:

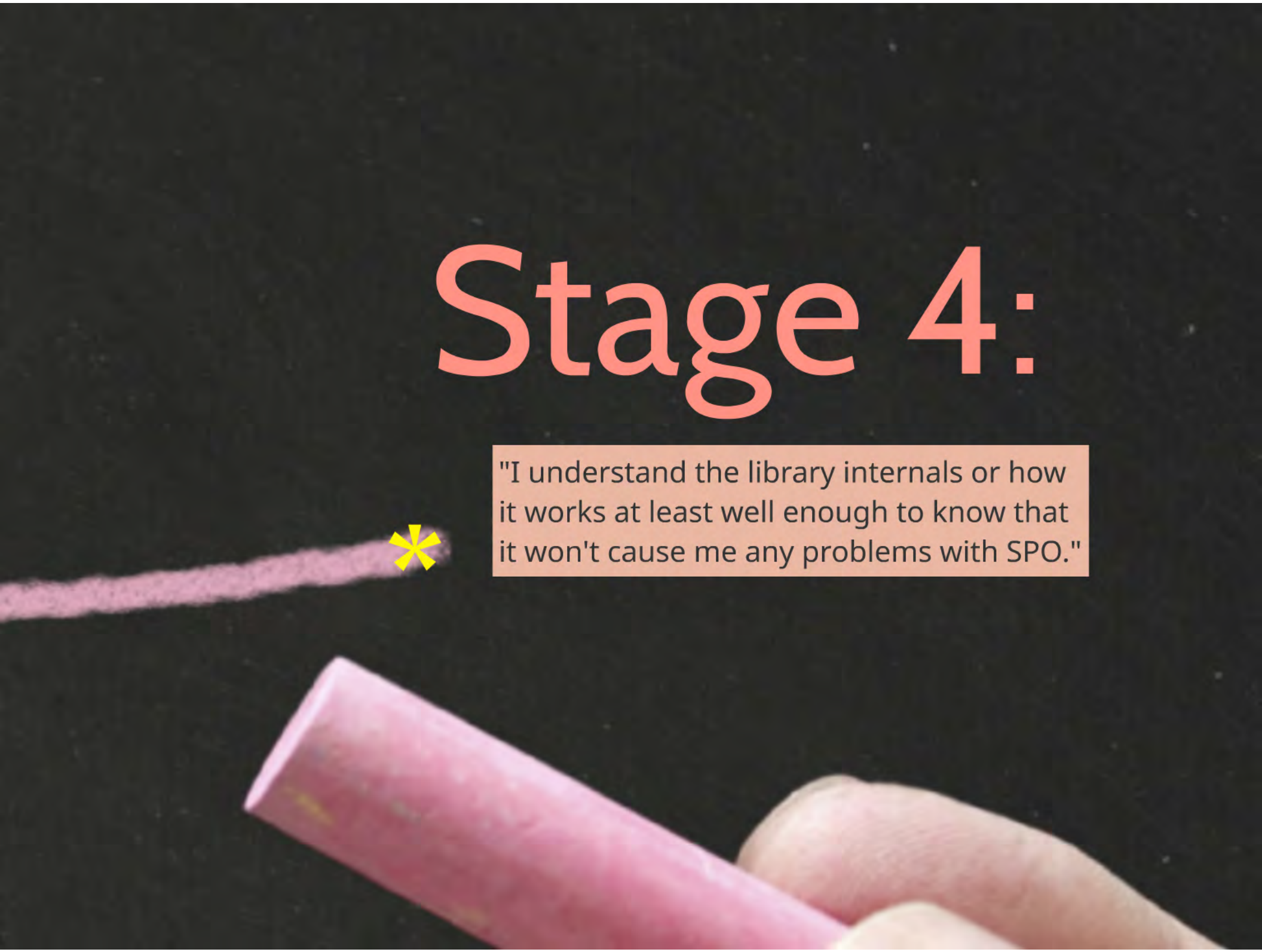


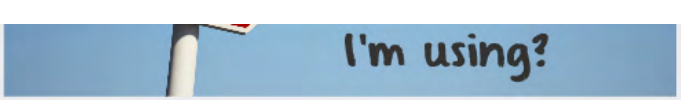
be illustrated with the following diagram of stages:



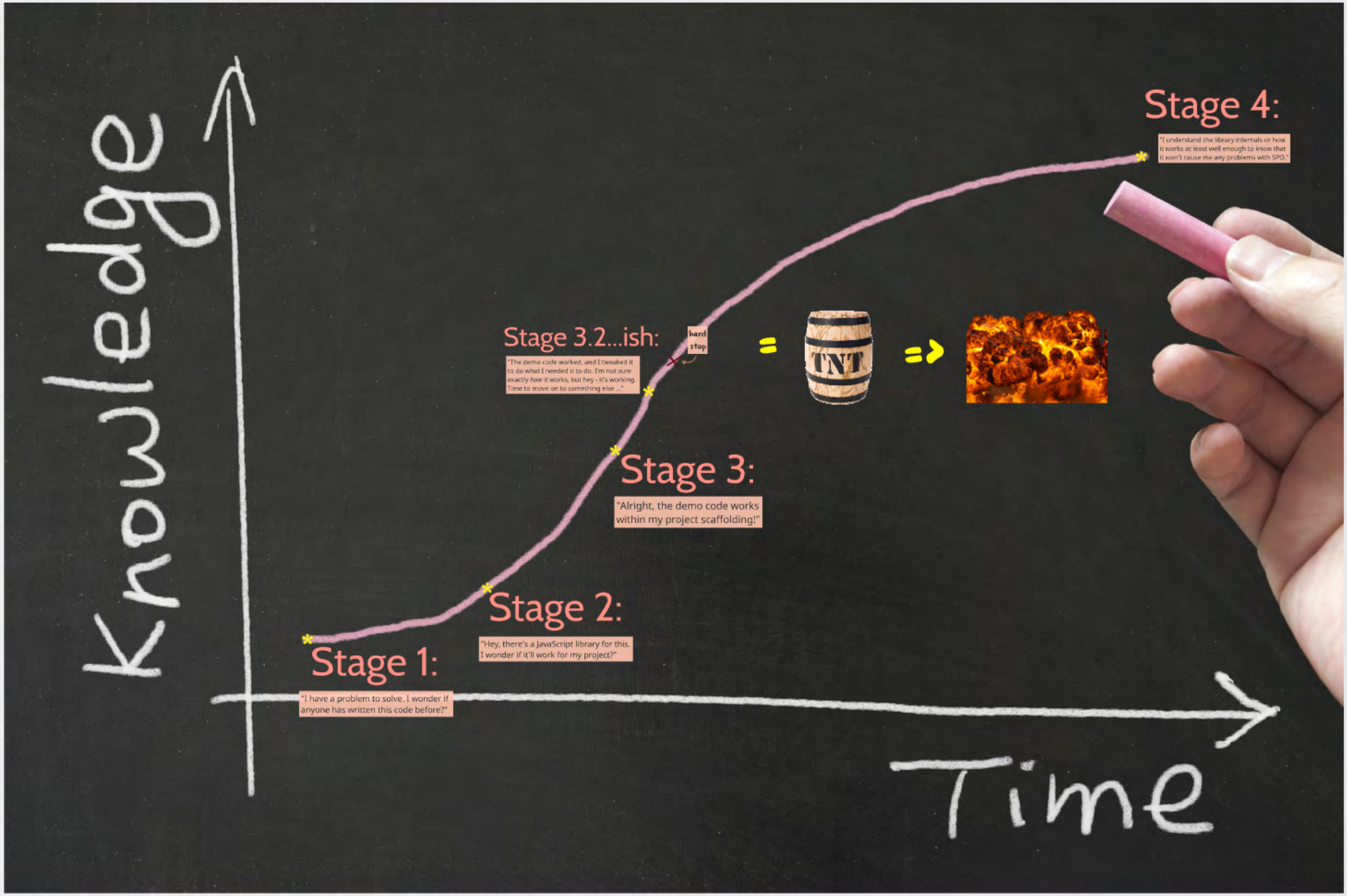

"Alright,
within m
*Stage 2:
"Hey, there's a JavaScript library for this.
I wonder if it'll work for my project?"
*Stage 1:
"I have a problem to solve. I wonder if
anyone has written this code before?"

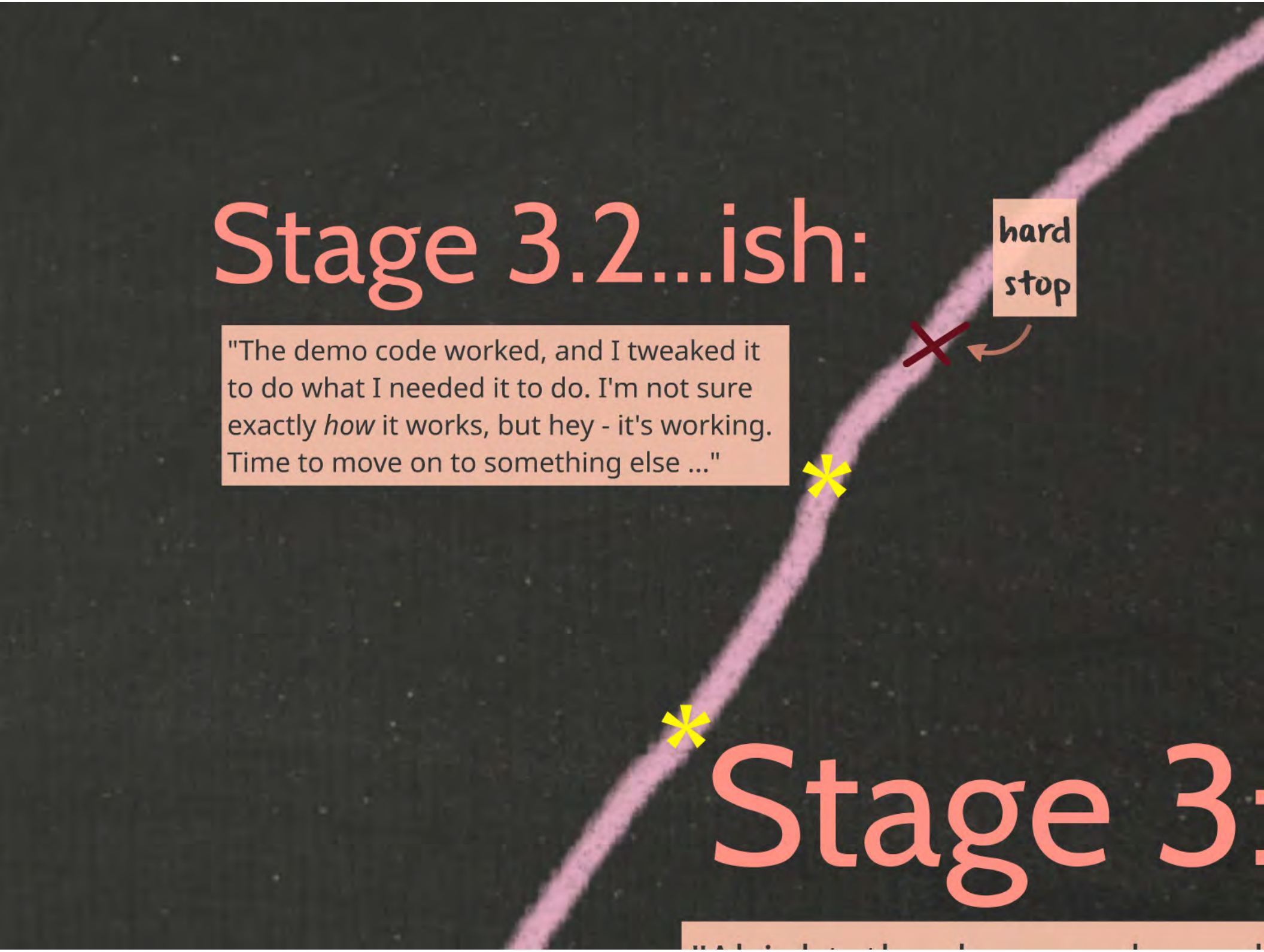

Stage 3:
"Alright, the demo code works within my project scaffolding!"

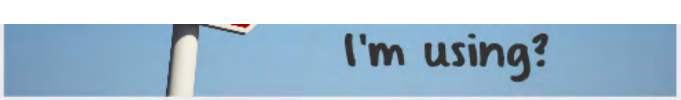

Stage 4:
"I understand the library internals or how it works at least well enough to know that it won't cause me any problems with SPO."
*



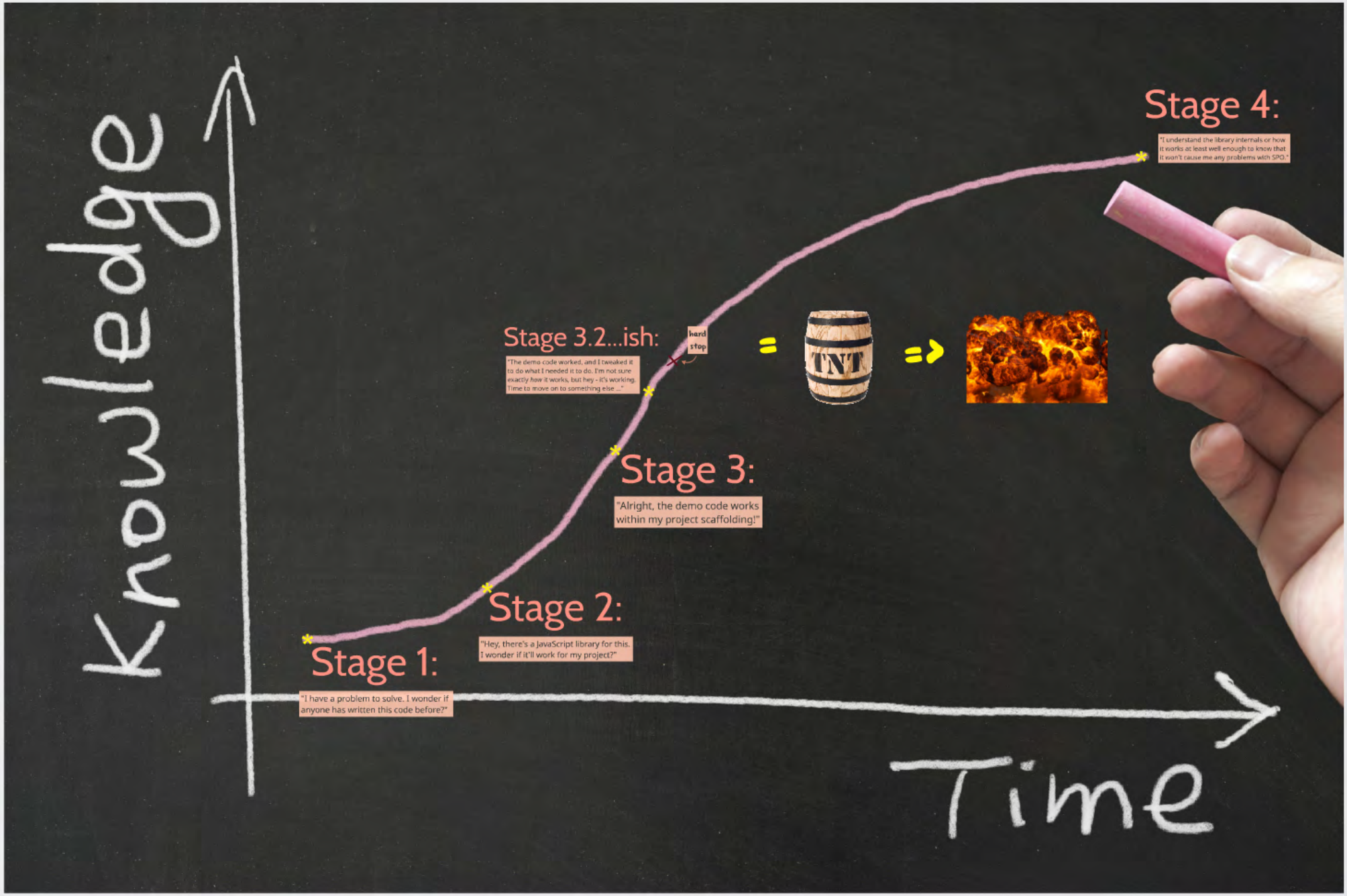
be illustrated with the following diagram of stages:




Stage 3.2...ish:
hard stop
"The demo code worked, and I tweaked it to do what I needed it to do. I'm not sure exactly *how* it works, but hey - it's working. Time to move on to something else ..."
Stage 3:



be illustrated with the following diagram of stages:



How many of you

How many of you know who this man is?

Chances are at least one or two of you have used code that he has created to get things done in your client-side development projects ...

Before CSOM/JSOM and REST
Marc's library simplified acces
for developers everywhere. H

Marc D. Anderson

- **creator of SPServices**
- master of client-side development and associated techniques

Before CSOM/JSOM and REST APIs - and before WCF SVC endpoints - Marc's library simplified access to the older ASMX web service endpoints for developers everywhere. It's still used heavily today.

Wow!

Last check on CodePlex (for SPServices):

Anderson

SPServices

client-side

> 47,000 downloads

So, getting back to "know your code/libraries" and how they work ...

As Marc will tell you, SPServices works just

So, getting back to "know your code/libraries" and how they work ...

As Marc will tell you, SPServices works just fine with SharePoint Online. But even Marc will tell you that you probably shouldn't use all of SPServices' methods when accessing SPO.

```
1   //Pre-populate all "Contact" fields with current user
2   var thisUserName = $().SPServices.SPGetCurrentUser({
3       fieldName: "Title",
4       debug: false
5   });
6   $().SPServices.SPFindPeoplePicker({
7       peoplePickerDisplayName: "Contact",
8       valueToSet: thisUserName,
9       checkNames: true
10  });
11  $().SPServices.SPFindPeoplePicker({
12      peoplePickerDisplayName: "Author/Contact",
13      valueToSet: thisUserName,
14      checkNames: true
15  });
16  $().SPServices.SPFindPeoplePicker({
17      peoplePickerDisplayName: "Organizer/Contact",
18      valueToSet: thisUserName,
19      checkNames: true
20  });
```

Consider this code.

It works just fine and does exactly what the comment indicates.

But it has a big problem.

anyone ever used the SPServices.SPGetCurrentUser() met

```
1   //Pre-populate all "Contact" fields with current user
2   var thisUserName = $().SPServices.SPGetCurrentUser({
3       fieldName: "Title",
4       debug: false
5   });
6   $().SPServices.SPFindPeoplePicker({
7       peoplePickerDisplayName: "Contact",
8       valueToSet: thisUserName,
9       checkNames: true
10  });
11  $().SPServices.SPFindPeoplePicker({
12      peoplePickerDisplayName: "Author/Contact",
13      valueToSet: thisUserName,
14      checkNames: true
15  });
16  $().SPServices.SPFindPeoplePicker({
17      peoplePickerDisplayName: "Organizer/Contact",
18      valueToSet: thisUserName,
19      checkNames: true
20  });
```

Consider this code.

It works just fine and does exactly what the comment indicates.

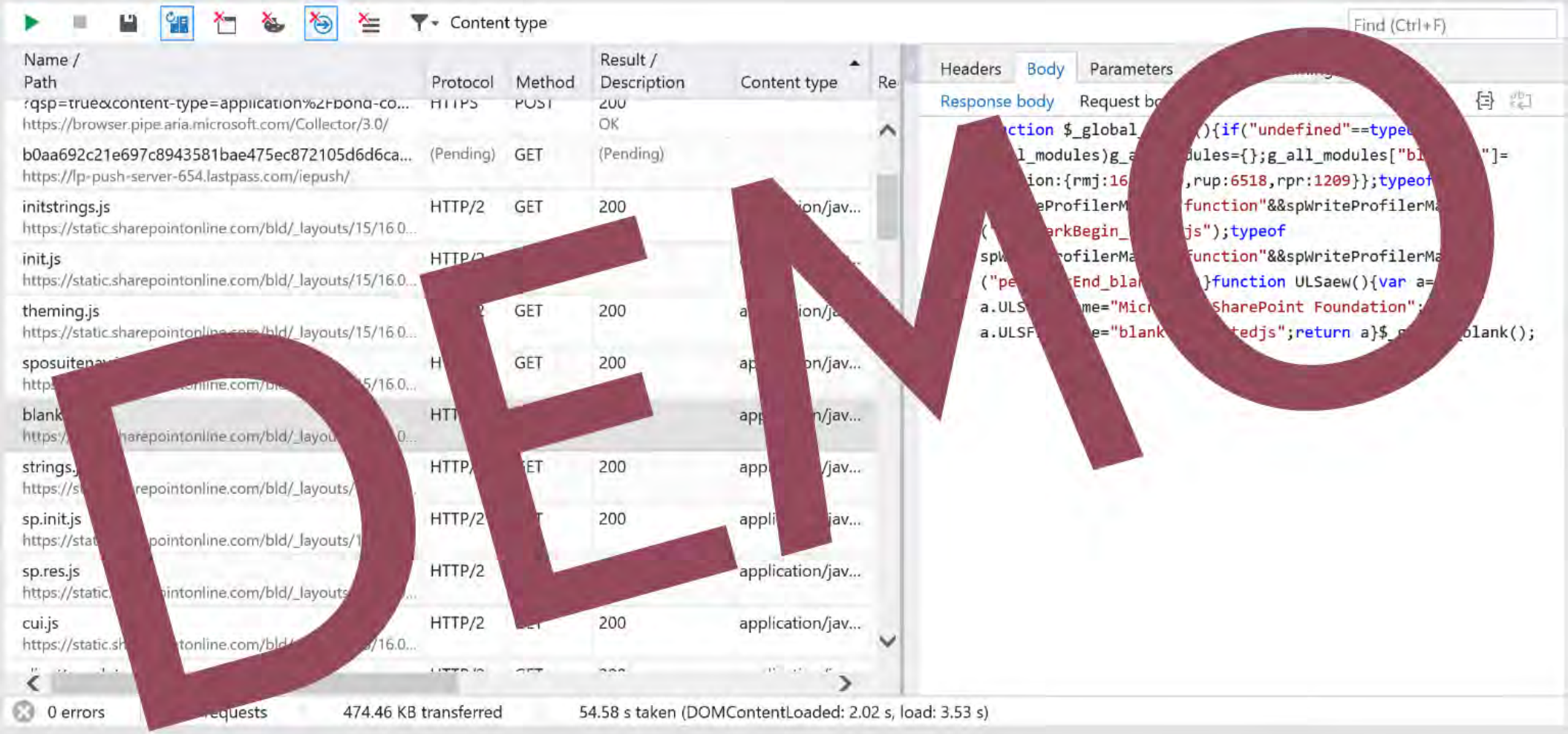
But it has a big problem.

Has anyone ever used the SPServices.SPGetCurrentUser() method?

Switching over to REST-based calls

WANTED: MARC ANDERSON

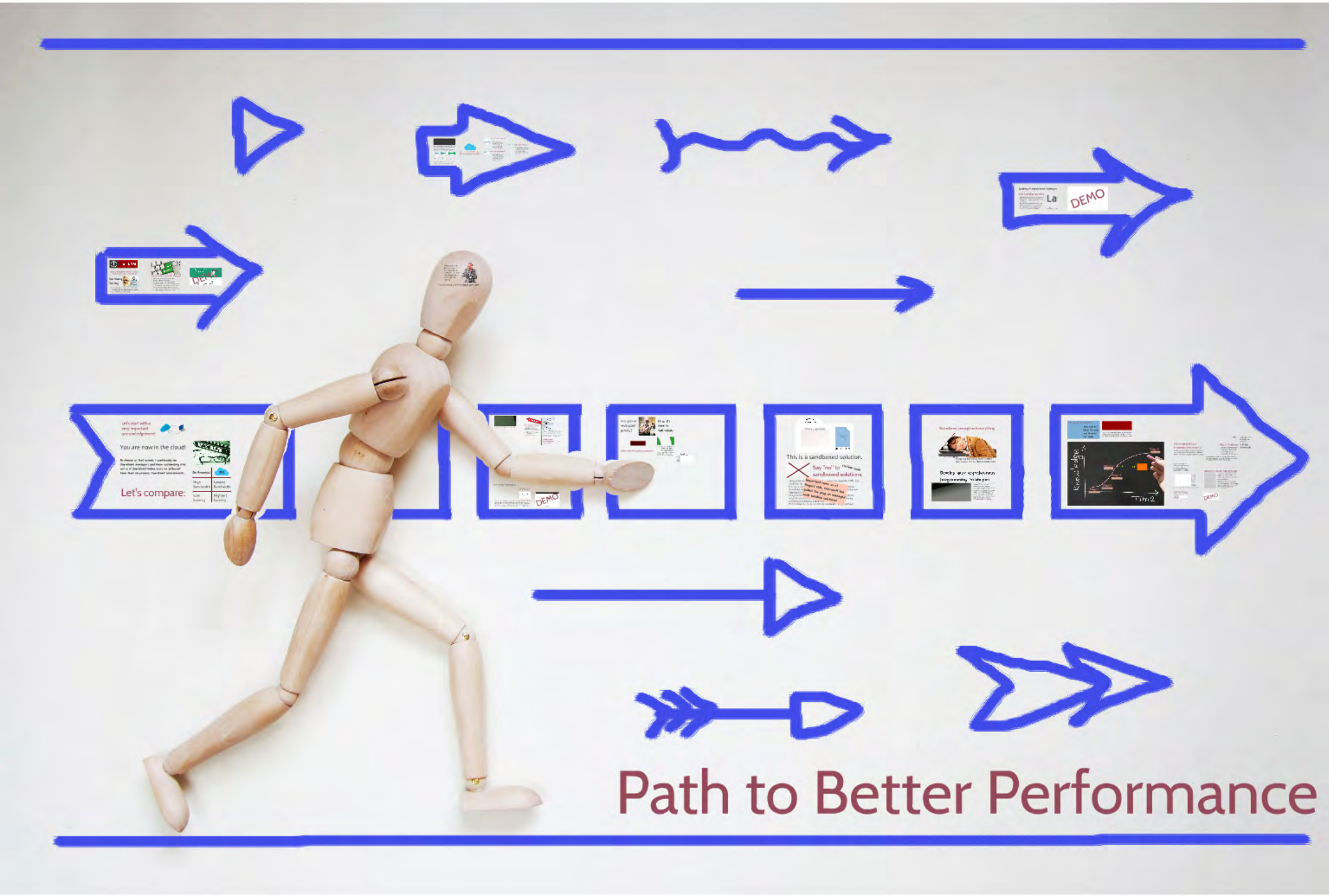
- Under the hood, SPGetCurrentUser() is generating an additional call to /_layouts/userdisp.aspx to “scrape” the contents of the page that is returned.
- If you (innocently) use SPGetCurrentUser() in your JavaScript files (especially multiple times in the context of a single page), you're creating all sorts of additional load on SPO and delaying the final results of your executing scripts.


Content type
Find (Ctrl+F)
Name / Path
Protocol
Method
Result / Description
Content type
Headers
Body
Parameters
Response body
Request bo
https://browser.pipe.aria.microsoft.com/Collector/3.0/
OK
b0aa692c21e697c8943581bae475ec872105d6d6ca...
https://lp-push-server-654.lastpass.com/iepush/
(Pending)
GET
(Pending)
initstrings.js
https://static.sharepointonline.com/bld/_layouts/15/16.0...
HTTP/2
GET
200
init.js
theming.js
sp.init.js
sp.res.js
cui.js
application/jav...
0 errors
474.46 KB transferred
54.58 s taken (DOMContentLoaded: 2.02 s, load: 3.53 s)
DEMO

Switching over to REST-based calls to get current user information can dramatically reduce execution time.

We had a script where SPGetCurrentUser() was being called several times. The results from swapping-in REST-based calls for the SPGetCurrentUser() calls:

```
I also performed some basic tests to capture the speed
differences. I performed each test 10 times and here are
the results:

*   Without the fix or browser caching - avg. 14.47 seconds
*   With the fix without browser caching - avg. 7.17 seconds
*   With the fix and browser caching - avg. 5.84 seconds
```


DEMO
Let's compare:
You are now in the cloud!
This is a sandboxed solution.
Say 'no' to sandboxed solutions.
DEMO
Path to Better Performance

Part 1:Quick Summary

- Don't treat SPO like your on-premises SharePoint farm. The two operate differently.
- Server-based caching isn't your friend (generally speaking) in SPO.
- Your browser can be your best friend when trying to troubleshoot SPO performance issues.
- Know the code you implement - or at least profile it before release.

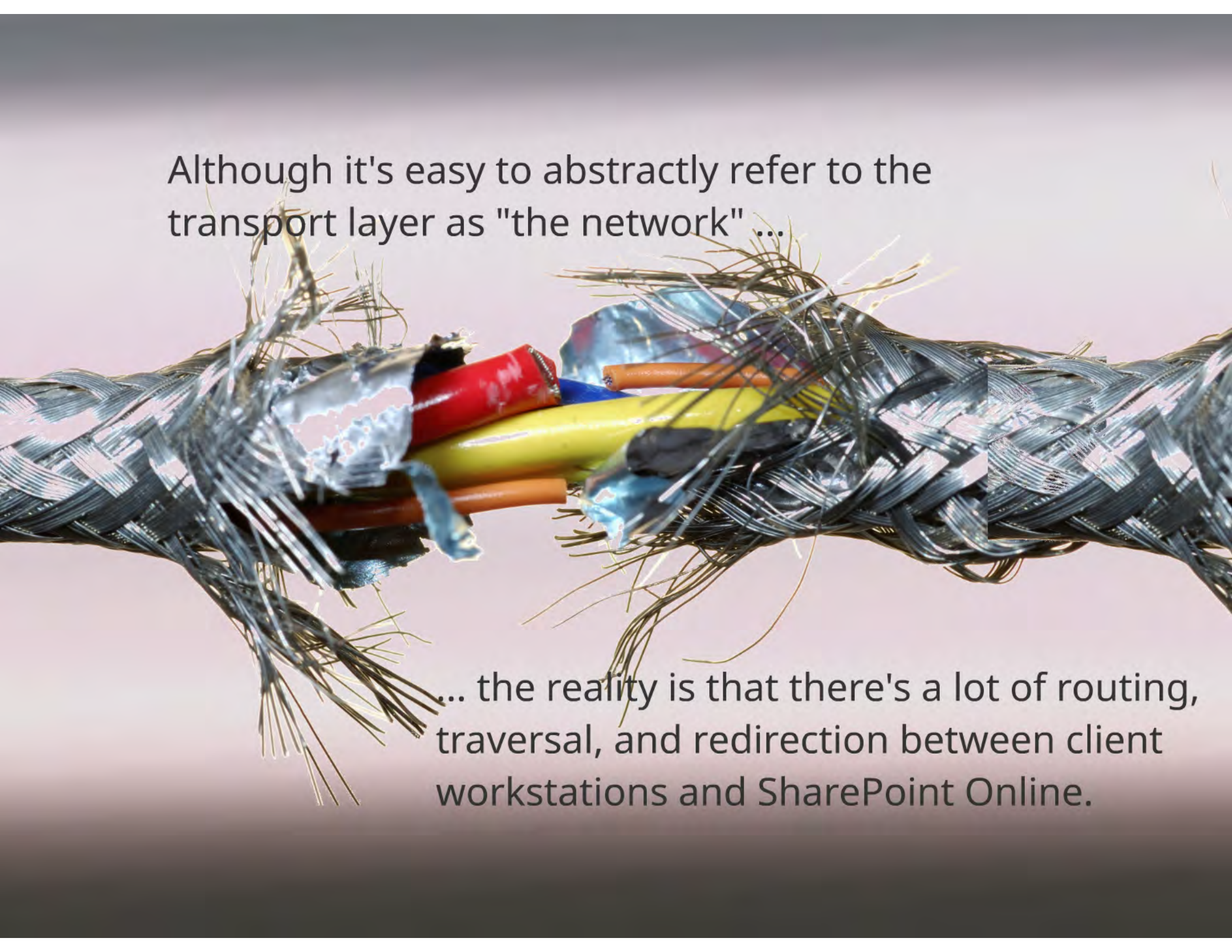

The Reality of Plumbing


The Network

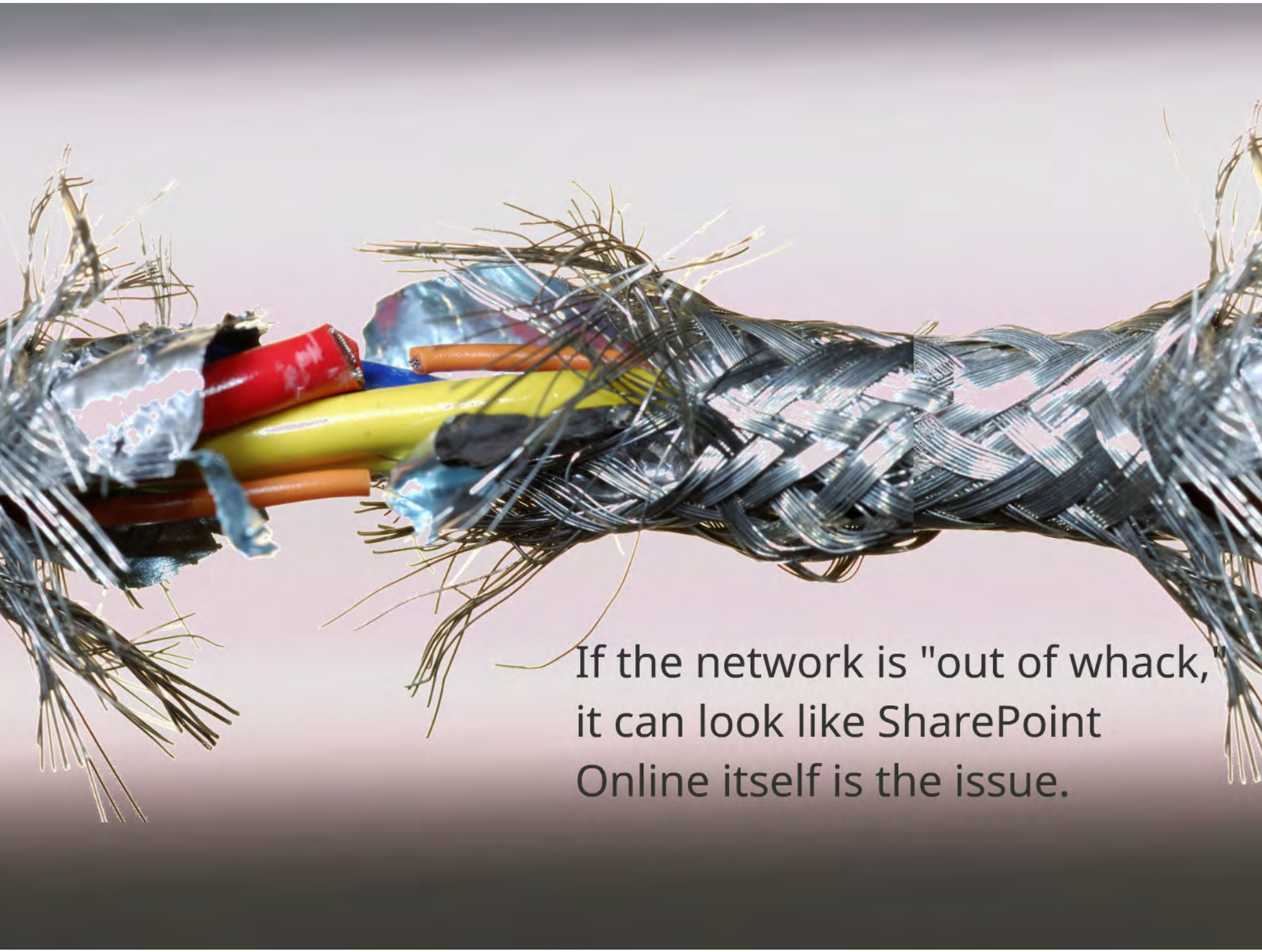
The Network

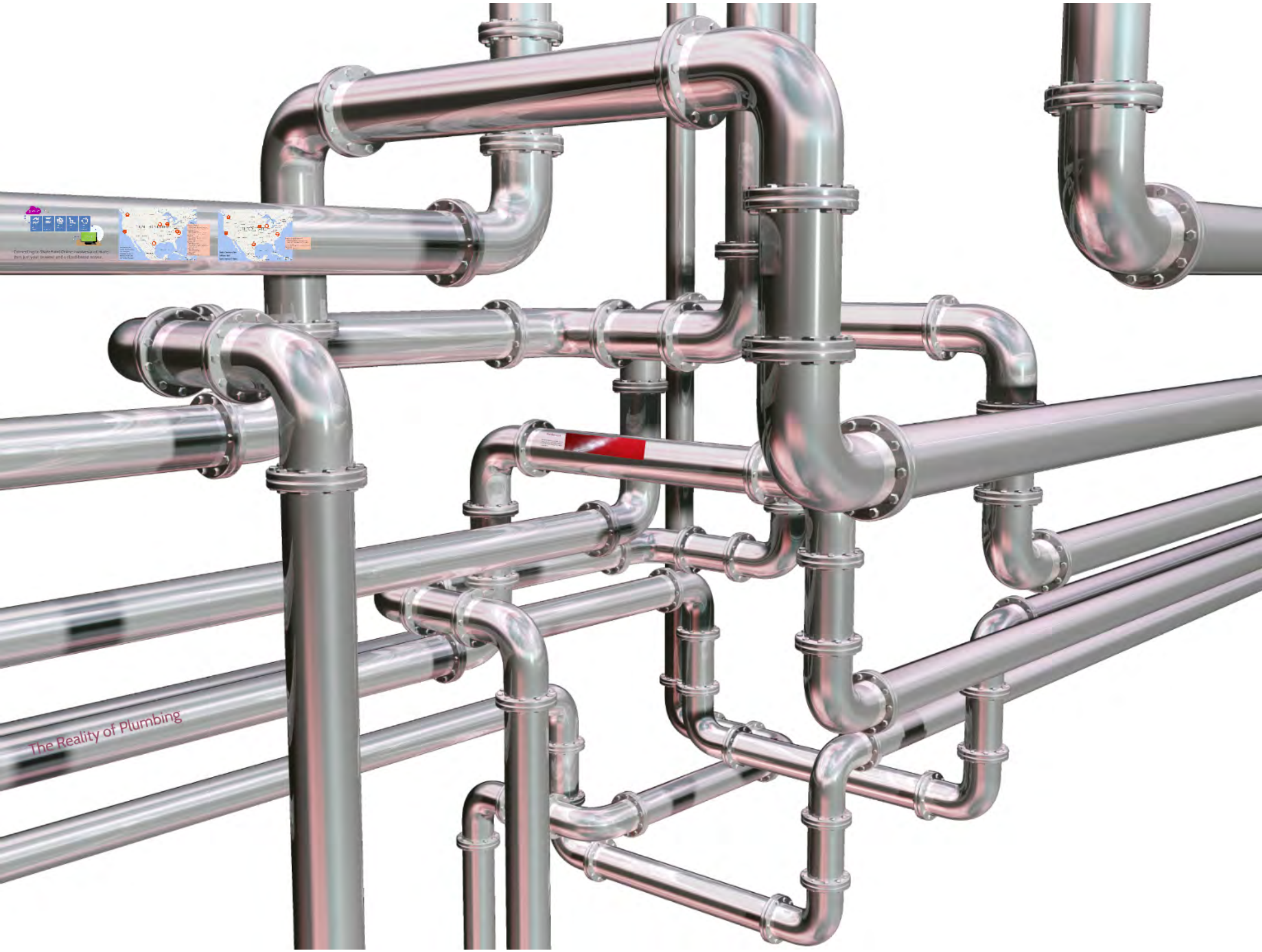
If we're talking about end-to-end performance, we really need to include the plumbing that carries everything.

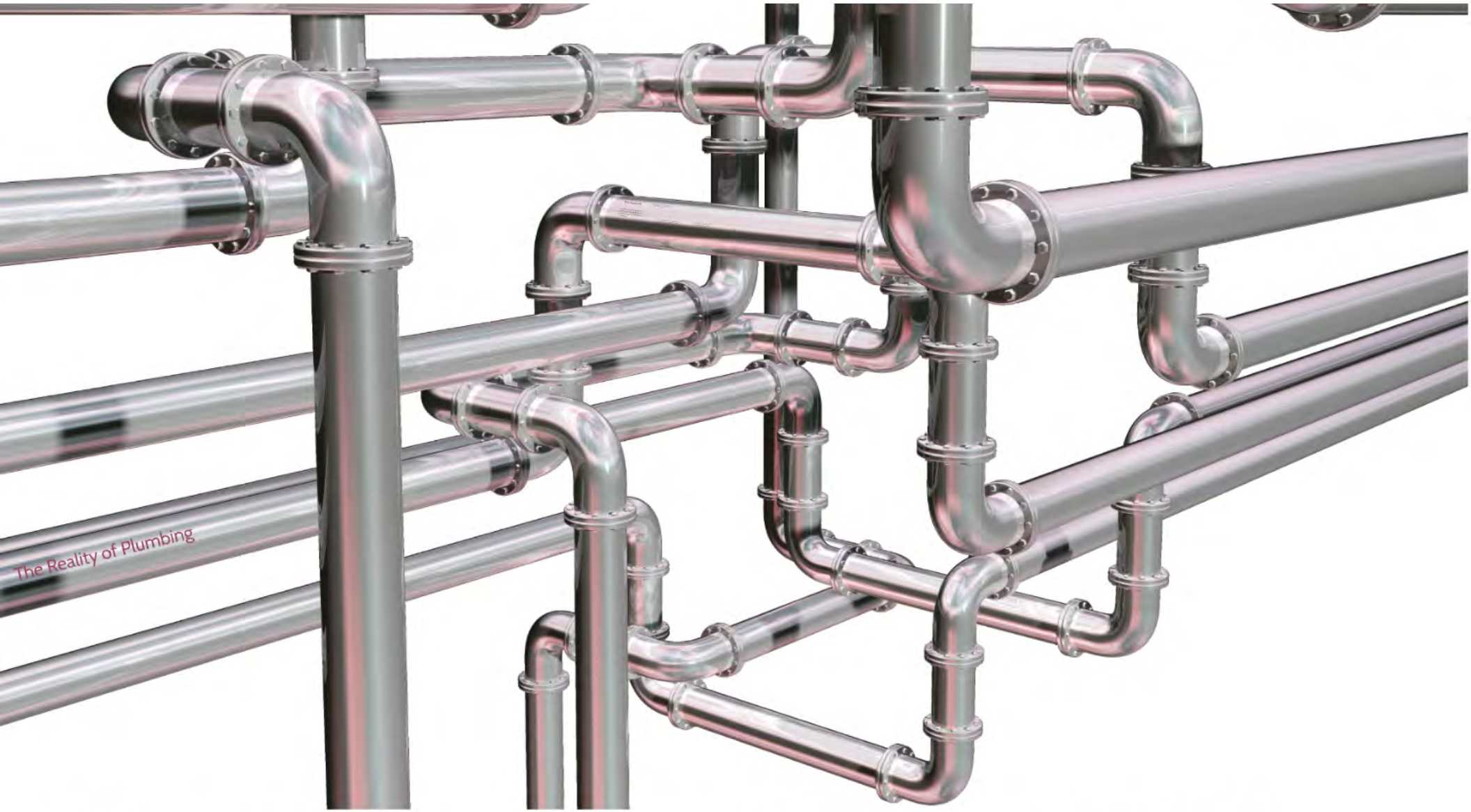
Although it's easy to abstractly refer to the transport layer as "the network" ...



... the reality is that there's a lot of routing, traversal, and redirection between client workstations and SharePoint Online.


If the network is "out of whack,"
it can look like SharePoint
Online itself is the issue.


The Reality of Plumbing

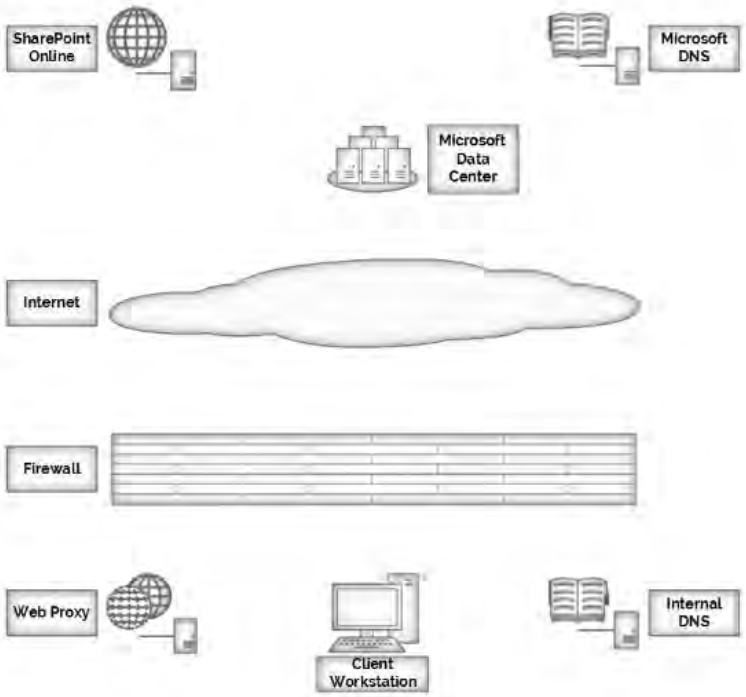


For some of you, your network connectivity to SPO may (unfortunately) look something like this.

So if you've done a lot of troubleshooting in SPO to no effect, it's time to consider the network.

Talking to SharePoint Online

- When a page or asset is requested from SharePoint Online, a series of conversations take place between the client workstation, internal servers, and Microsoft's servers.
- The image on the right highlights the major participants in the conversations.



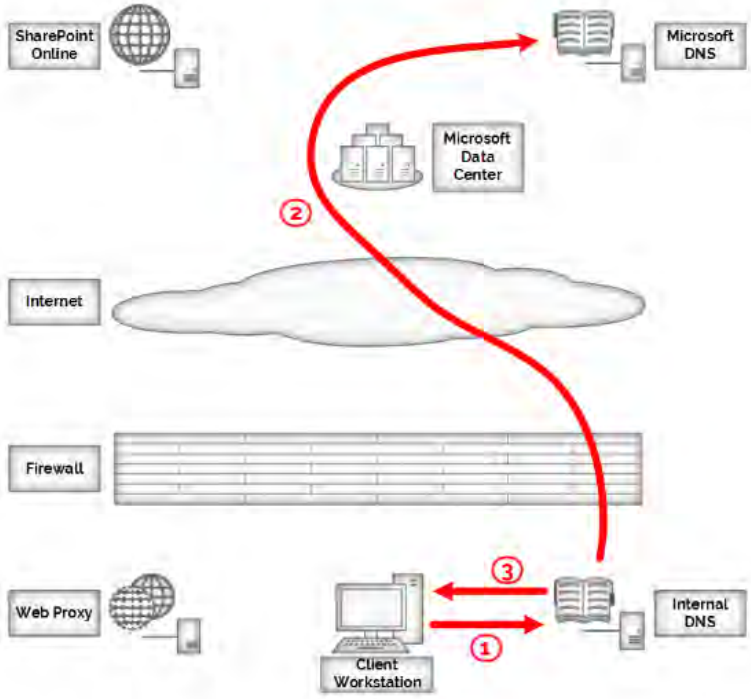
Let's say you wanted to retrieve a page from SPO ...

The first step involves you client workstation resolving the hostname

the major participants in the conversations.

Let's say you wanted to retrieve a page from SPO ...

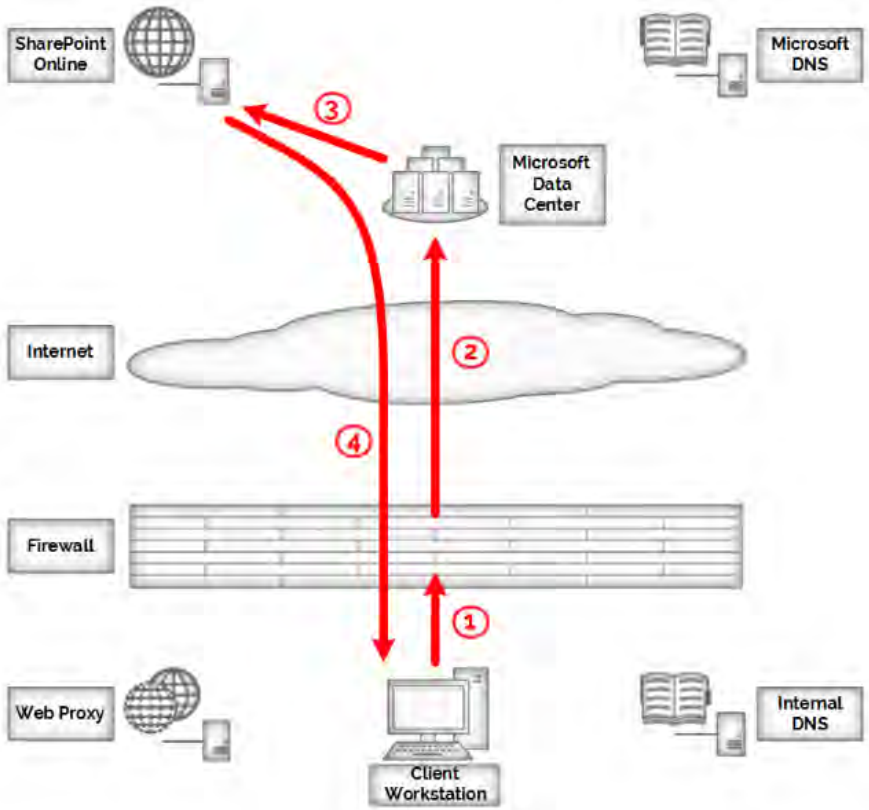
The first step involves you client workstation resolving the hostname (e.g., myspo.sharepoint.com) to its IP address.



1. The workstation contacts an internal DNS server to ask for the appropriate IP address.
2. If the internal DNS server doesn't have the IP address for the hostname cached, it passes the lookup up-the-chain to Microsoft's DNS server.
3. Microsoft's DNS server responds with the IP address the client workstation needs.

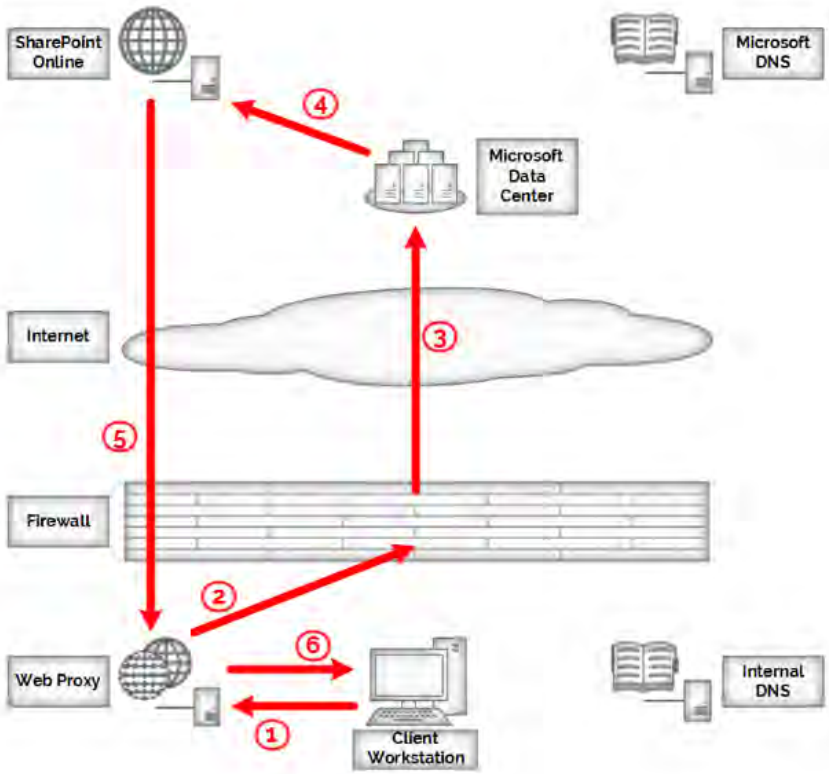
Step two involves making the actual request of the SharePoint Online server.

1. Armed with the appropriate IP address, the client workstation issues an HTTP request to the SharePoint Online Server. That request travels the local network to the firewall or gateway.
2. In common scenarios, the firewall performs network address translation (NAT) to pass the internal client request on to the public Microsoft network entry point.
3. Within the Microsoft network, the request makes its way to the SharePoint Online servers for the client's tenant.
4. An SPO front-end serves up the request.



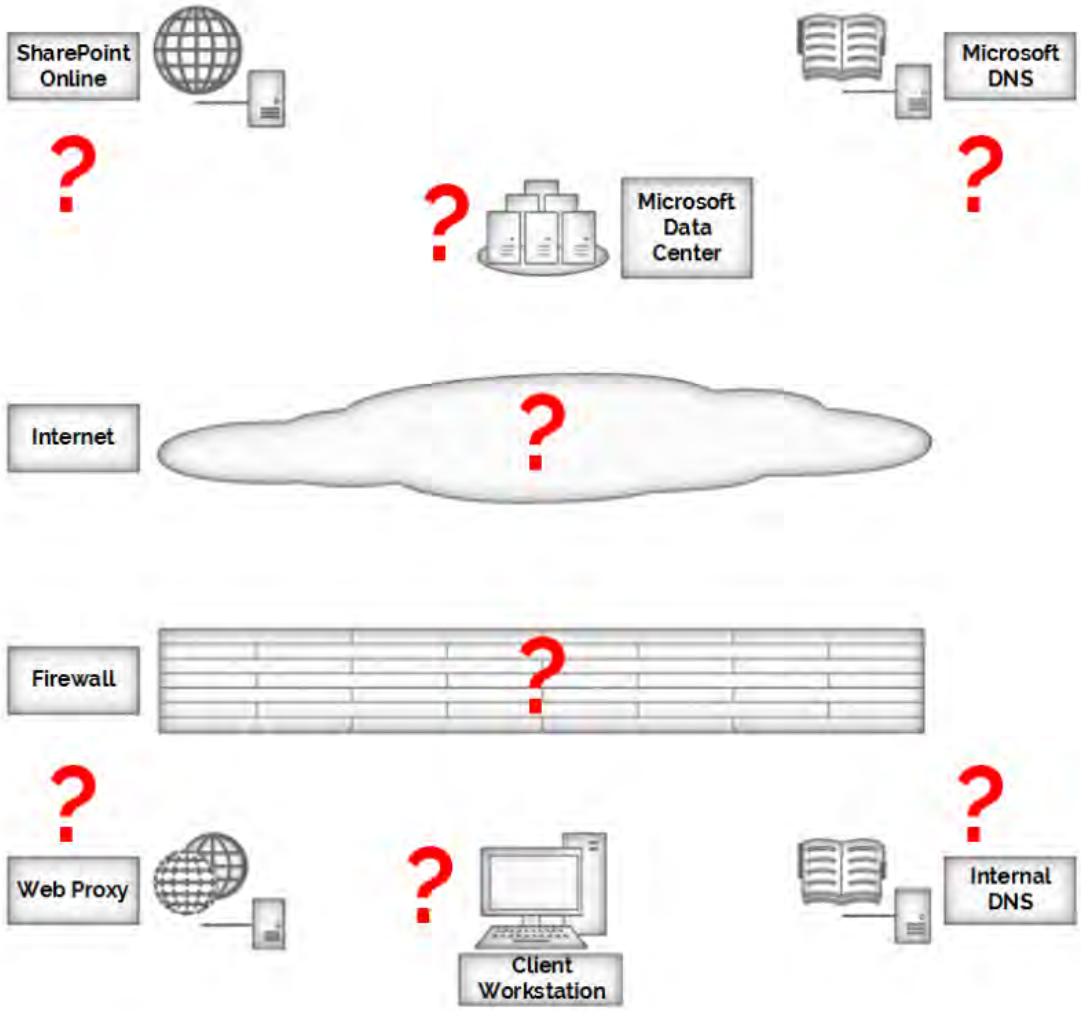
In enterprise environments, the SPO request may take a slightly different path.

1. Rather than going directly to a firewall or gateway, the client's request goes to a web proxy.
2. The web proxy makes the actual outbound request.
3. In common scenarios, the firewall performs network address translation (NAT) to pass the proxy's call through.
4. Within the Microsoft network, the request makes its way to the SharePoint Online servers for the client's tenant.
5. An SPO front-end serves up the request.
6. The web proxy passes the response back to the client.



6. The web proxy passes the response back to the client.

What could possibly go wrong?



Network Troubleshooting and Tools

The basics (in sweeping generalities)

- When adopting Office 365, Microsoft recommends that you start with a minimum of 20% headroom on your network circuits.
- SharePoint Online's network load consists of two main components
  - HTTPS page requests
  - List/library uploads and downloads
- You may have seen the matrix on the right. It is merely a starting point.

e

rent 0%)	Central solution	Dist
	Bandwidth: 3Mbps (dual T1) Latency: less than 100 ms	Banc Later
	Bandwidth: more than 3Mbps (dual T1) Latency: less than 250 ms	Banc Later
	Bandwidth: more than 3Mbps (dual T1) Latency: less than 250 ms	Banc Later

The basics (in sweeping generalities)

- In order to improve your understanding of bandwidth requirements, you need to establish a baseline. Don't just wing-it with "educated" guesses on bandwidth usage!
- In reality, SharePoint Online communications tend to occur is bursts … and they're fairly latency tolerant.
- Something to remember (as an aside): OneDrive for Business (OD4B) leverages SharePoint MySites for document storage.

SharePoint Online

Number of users	Concurrent users (10%)	Central solution	Distributed solution
100-5,000	10-500	Bandwidth: 3Mbps (dual T1) Latency: less than 100 ms	Bandwidth: 1.5Mbps (T1) Latency: less than 100 ms
10,000	1,000	Bandwidth: more than 3Mbps (dual T1) Latency: less than 250 ms	Bandwidth: 1.5 Mbps (T1) Latency: less than 250 ms
100,000	10,000	Bandwidth: more than 3Mbps (dual T1) Latency: less than 250 ms	Bandwidth: 1.5 Mbps (T1) Latency: less than 250 ms

Something's not right!

- You've got more than enough bandwidth to cover SharePoint Online connections, you don't think that you're suffering from any latency problems, but users are still complaining of poor SPO performance.
- Where do you begin?

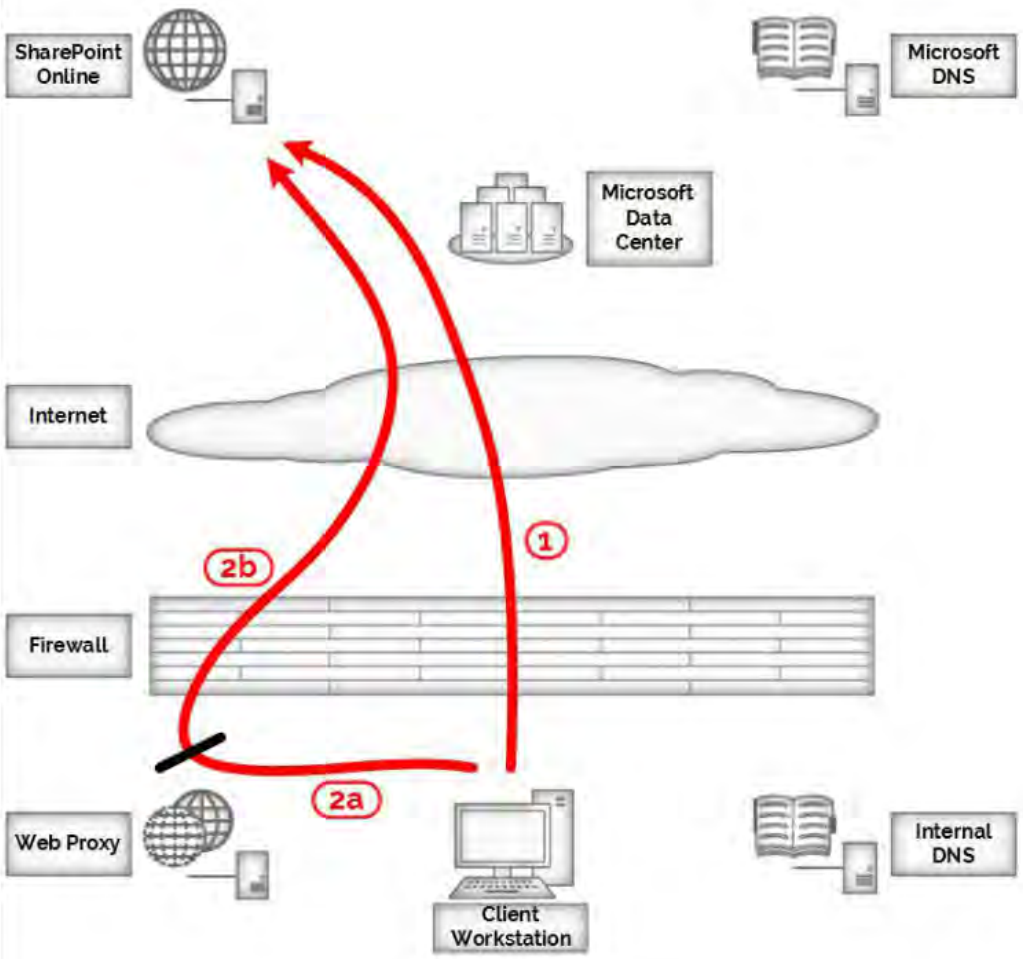
Note: no servers were actually harmed in the capture of the picture on the right.

Understand your network and egress

- In all likelihood, the problem is not with Microsoft's network.
- Start by ensuring that you understand the path that requests take to get from a client workstation to Office 365
  - Do requests traverse a firewall? In all likelihood, the answer is "yes."
  - Is a proxy server part of the request path/sequence?
  - What else sits on the internal network between client workstations and the Internet?

Start simple. Validate latency.

- Latency is the delay between sending a request (e.g., for a SharePoint page) and receiving a response.
- A ping's round-trip time (RTT) is measured in milliseconds (ms).
- If you have a web proxy, you need to measure latency between workstation and proxy (2a) as well as proxy and SPO (2b).
- Use a "firewall-friendly" ping tool like PSPing from SysInternals



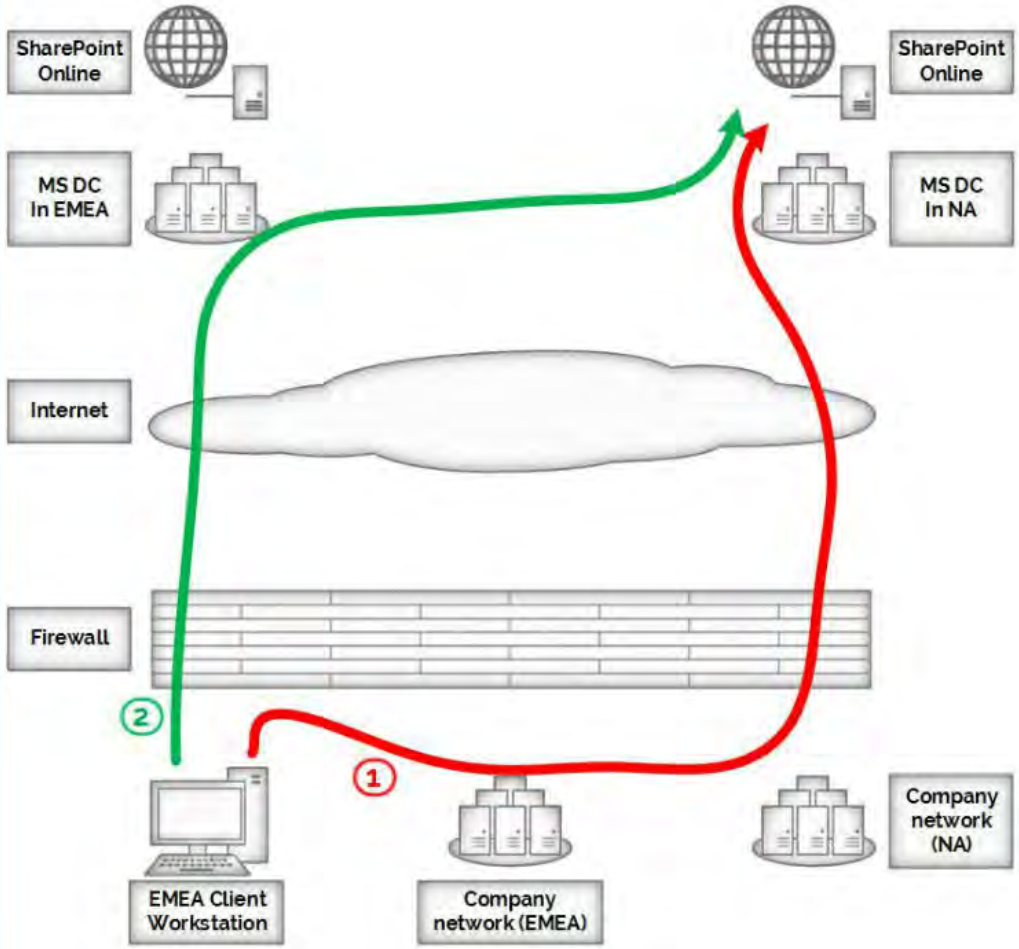
Demo: PSPing

- Consider your RTT value (or values) in the context of your network topology.
- With a straight connection in-region, sub-100ms times are the norm.
- If you are part of a global organization with centralized egress out-of-region, RTT to your proxy may be high.
  - Subtract RTT-to-proxy from RTT-to-SPO; the result is your RTT from proxy to SPO
- High values outside your network? Inspect from the egress point, seek ISP assistance.

```
PS C:\SysInternals> .\psping.exe -n 10 cardinalsolutions.sharepoint.com:443

PsPing v2.01 - PsPing - ping, latency, bandwidth measurement utility
Copyright (C) 2012-2014 Mark Russinovich
Sysinternals - www.sysinternals.com

TCP connect to 191.234.214.34:443:
11 iterations (warmup 1) connecting test:
Connecting to 191.234.214.34:443 (warmup): 35.07ms
Connecting to 191.234.214.34:443: 35.52ms
Connecting to 191.234.214.34:443: 36.35ms
Connecting to 191.234.214.34:443: 33.93ms
Connecting to 191.234.214.34:443: 34.53ms
Connecting to 191.234.214.34:443: 34.35ms
Connecting to 191.234.214.34:443: 35.32ms
Connecting to 191.234.214.34:443: 31.62ms
Connecting to 191.234.214.34:443: 33.49ms
Connecting to 191.234.214.34:443: 34.40ms
Connecting to 191.234.214.34:443: 35.15ms

TCP connect statistics for 191.234.214.34:443:
  Sent = 10, Received = 10, Lost = 0 (0% loss),
  Minimum = 31.62ms, Maximum = 36.35ms, Average = 34.47ms
PS C:\SysInternals>
```

Are your DNS lookups "good?"

- The goal is to get your traffic onto the Microsoft backbone with as few network hops as possible.
- If you work in an organization with multiple data centers, determine if your web traffic goes out within your region.
- If your requests are being proxied in (or routed to) a different region (line #1 on the right), your requests may be making a lot of needless and/or constrained network hops. Your goal should be a path like the one described by line #2.



Demo: DNS checks

- With a simple test, you can determine if your company's DNS lookups are returning results that get your traffic onto Microsoft's backbone in your region.
- Easiest way? Perform a route trace (tracert) to outlook.office365.com and note the hostname component of the result.
- Are you seeing a name that's out of region? Engage the network team to get DNS lookups performed in-region rather than centrally – at least for Office 365 hosts.

```
PS C:\SysInternals> tracert outlook.office365.com

Tracing route to outlook-namnorth.office365.com [132.245.71.178]
over a maximum of 30 hops:
```

```
PS C:\SysInternals> tracert sharepoint.com
```

Routing and peering issues

- If DNS doesn't seem to be the issue, you may still more hops than desirable before traffic reaches the Microsoft network from your own network egress point.
- Use tracert (if ICMP – Internet Control Message Protocol – is open) or TraceTCP to identify the route to SPO.
- Find the entry point to the Microsoft network (msn.net) in the results. If a large number of hops (or expensive hops) happen first, you may have an issue.

```
PS C:\SysInternals> tracert sharepoint.com

Tracing route to sharepoint.microsoft.com [64.4.6.100]
over a maximum of 30 hops:

  1     *        *        *     Request timed out.
  2     5 ms     4 ms     5 ms  29.242.117.66.ded-dsl.fuse.net [66.117.242.29]
  3     6 ms     6 ms     5 ms  nw1-dsl-72-49-231-1.fuse.net [72.49.231.1]
  4     6 ms     7 ms     7 ms  216.68.14.102
  5     6 ms     7 ms     7 ms  216.68.14.101
  6    12 ms    13 ms    14 ms  216.68.14.85
  7    13 ms    13 ms    13 ms  chi-8075.msn.net [206.223.119.27]
  8    17 ms    15 ms    15 ms  ae4-0.ch1-96c-2b.ntwk.msn.net [104.44.224.90]
  9    68 ms    69 ms    68 ms  ae-74.ibr02.ch1.network.microsoft.com [104.44.8.42]
 10    72 ms    70 ms    69 ms  be-5-0.ibr01.dm2.network.microsoft.com [104.44.4.76]
 11    71 ms    69 ms    69 ms  be-3-0.ibr01.dm3.network.microsoft.com [104.44.4.74]
 12    69 ms    71 ms    69 ms  be-4-0.ibr02.den07.network.microsoft.com [104.44.4.97]
 13    71 ms    70 ms    71 ms  be-1-0.ibr01.den07.network.microsoft.com [104.44.4.58]
 14    69 ms    69 ms    70 ms  be-2-0.ibr01.dna.network.microsoft.com [104.44.4.108]
 15    68 ms    68 ms    69 ms  be-5-0.ibr02.bay.network.microsoft.com [104.44.4.120]
 16    67 ms    68 ms    68 ms  ae72-0.bay-96c-1b.ntwk.msn.net [104.44.8.179]
 17    68 ms    68 ms    68 ms  10.22.40.13
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```

Other considerations (for the network folks)

- Check your TCP window scaling setting on devices. Is it enabled? If not, you may be multiplying the effects of latency.
- Do you have a large number of users connecting to SharePoint online and/or other Office 365 workloads? Due to NAT limitations, a firewall or similar device can support a maximum of 6,000 clients per single public IP address. If you exceed this, connectivity to Office 365 and SPO may be intermittent and unpredictable.

Other considerations (for the network folks)

- Is the performance between clients and SPO highly variable and/or unpredictable? Perhaps there is a packet loss issue. If TCP retransmits are high (usually > 1%), then it suggests a connectivity issue somewhere.
- Are your connections completing slowly? Do you have a web proxy? If the proxy is requiring authentication for Office 365 and SPO traffic, then it might be slowing connections down needlessly. Microsoft recommends disabling proxy authentication for Office 365 and SPO.

Other considerations (for the network folks)

- How's your DNS lookup performance? If you flush your DNS cache and do a lookup for your SharePoint site – and it takes longer than milliseconds (e.g., two or three seconds), then you may have a DNS issue somewhere in the lookup chain.
- Check the TCP options on both the SYN and the SYN/ACK packets being sent:
  - MSS (MaxSegmentSize) should be near the max of 1460 and not much lower.
  - SACK (selective acknowledgement) should be enabled to handle dropped packets efficiently.


!&A

References

SPTechCon Austin 2016
http://www.sptechcon.com/

“Technical diagrams for SharePoint 2013”
https://technet.microsoft.com/en-us/library/cc263199.aspx

“Office 365 Performance Management”
https://mva.microsoft.com/en-US/training-courses/office-365-performance-management-8416

References

“Where is my data?”
https://www.microsoft.com/online/legal/v2/en-us/MOS_PTC_Geo_Boundaries.htm

“Deploying Office 365”
http://video.ch9.ms/sessions/project/2014/PC215_Medford.pptx

“New datacenter regions for Office 365”
https://support.office.com/en-us/article/Network-and-migration-planning-for-Office-365-f5ee6c33-bcd7-4b0b-b0f8-dc1d9fb8d132?ui=en-US&rs=en-US&ad=US#calculators

References

“Network and migration planning for Office 365”
https://support.office.com/en-us/article/Network-and-migration-planning-for-Office-365-f5ee6c33-bcd7-4b0b-b0f8-dc1d9fb8d132?ui=en-US&rs=en-US&ad=US#calculators

Fiddler web debugging proxy
http://www.telerik.com/fiddler

“FAQ – Certificates in Fiddler”
http://www.telerik.com/blogs/faq---certificates-in-fiddler

References

“Analyzing your webpage’s network traffic”
https://msdn.microsoft.com/en-us/library/dn255004(v=vs.85).aspx

“Diagnosing performance issues with SharePoint Online”
https://support.office.com/en-us/article/Diagnosing-performance-issues-with-SharePoint-Online-3c364f9e-b9f6-4da4-a792-c8e8c8cd2e86

“Image optimization for SharePoint Online”
https://support.office.com/en-gb/article/Image-optimization-for-SharePoint-Online-c7edb02a-fdab-4f91-9a20-cba01dad28ef

References

Font Awesome
https://fortawesome.github.io/Font-Awesome/

“How to improve your website performance with a CDN”
http://b2evolution.net/web-hosting/blog/tips/improve-website-performance-with-cdn

“Navigation options for SharePoint Online”
https://support.office.com/en-us/article/Navigation-options-for-SharePoint-Online-adb92b80-b342-4ecb-99a1-da2a2b4782eb

References

“Using Content Search Web Part instead of Content Query Web Part to improve performance in SharePoint Online”

https://support.office.com/en-us/article/Using-Content-Search-Web-Part-instead-of-Content-Query-Web-Part-to-improve-performance-in-SharePoint-Online-e8ce6b72-745b-464a-85c7-cbf6eb53391b

“Using jQuery Promises & Deferreds with SharePoint 2013”

http://blog.qumsieh.ca/2013/10/31/using-jquery-promises-deferreds-with-sharepoint-2013-jsom/

References

“Lazy Load Plugin for jQuery”
https://github.com/tuupola/jquery_lazyload

“Documentation > $().SPServices.SPGetCurrentUser”
http://spservices.codeplex.com/wikipage?title=%24%28%29.SPServices.SPGetCurrentUser

“SPServices Example: UserProfileService.GetUserProfileByName”
http://sympmarc.com/2012/02/15/spservices-example-userprofileservice-getuserprofilebyname/

References

“get current user info using jquery, REST and csom”
http://sharepoint.stackexchange.com/questions/124909/get-current-user-info-using-jquery-rest-and-csom

“Users, groups and the roles REST API reference”
https://msdn.microsoft.com/en-us/library/office/dn531432.aspx#bk_User

"Caching, You Ain't No Friend Of Mine"
http://sharepointinterface.com/2016/01/31/caching-you-aint-no-friend-of-mine/

References

“Deprecation of Custom Code in Sandboxed Solutions"
https://blogs.msdn.microsoft.com/sharepointdev/2014/01/14/deprecation-of-custom-code-in-sandboxed-solutions/

“Using the Object Cache with SharePoint Online"
https://support.office.com/en-us/article/Using-the-object-cache-with-SharePoint-Online-38bc9c14-3826-449c-beb6-b1003bcbeaaf

References

“Network planning and performance tuning for Office 365"
https://support.office.com/en-us/article/Network-planning-and-performance-tuning-for-Office-365-e5f1228c-da3c-4654-bf16-d163daee8848?ui=en-US&rs=en-US&ad=US

“Network and migration planning for Office 365"
https://support.office.com/en-us/article/Network-and-migration-planning-for-Office-365-f5ee6c33-bcd7-4b0b-b0f8-dc1d9fb8d132?ui=en-US&rs=en-US&ad=US#netreference

References

“Top 10 Tips for Optimising & Troubleshooting your Office 365 Network Connectivity"
https://blogs.technet.microsoft.com/onthewire/2014/06/18/top-10-tips-for-optimising-troubleshooting-your-office-365-network-connectivity/

“Where is my data?"
https://www.microsoft.com/online/legal/v2/?docid=25

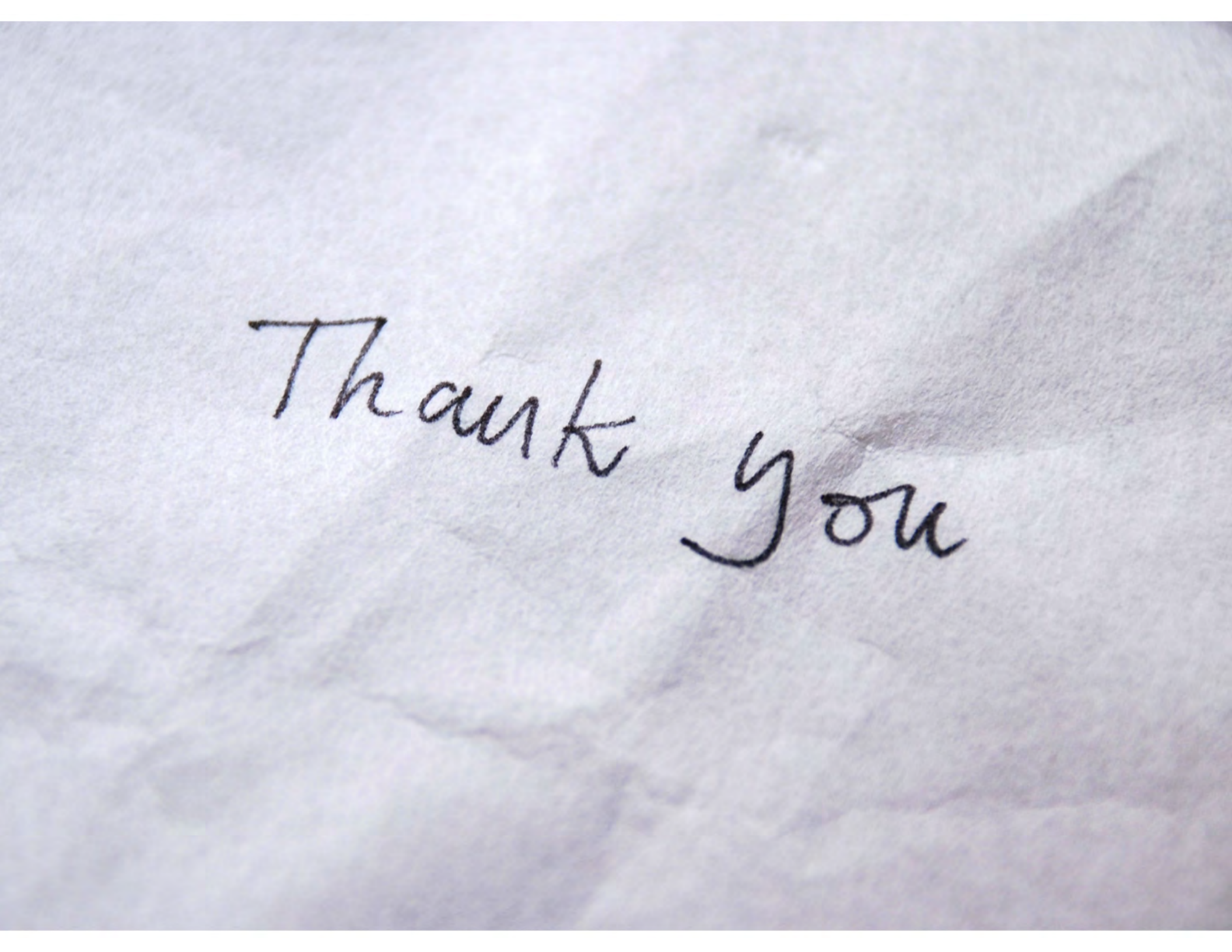
“SysInternals PSPing"
https://live.sysinternals.com/psping.exe

References

“TraceTCP on Github"
http://simulatedsimian.github.io/tracetcp.html

“WinPcap download (required for TraceTCP)"
https://www.winpcap.org/install/

“Preventing proxy authentication from delaying your O365 connection"
http://blogs.technet.com/b/onthewire/archive/2014/06/16/preventing-proxy-authentication-from-delaying-your-o365-connection.aspx


Thank you

Sean P. McDonough

SharePoint & Office 365 Gearhead, Tinkerer, and Microsoft MVP

bitstream FOUNDRY

Email: sean@bitstreamfoundry.com
Twitter: @spmdonough
Blog: http://SharePointInterface.com
About: http://about.me/spmcdonough