Welcome.
Congratulations for **12** Awesome Years!
CodeStock: Looking Back

@JeremyLikness
CodeStock 2008 Developer's Conference

CodeStock's mission is to bring the best and brightest code experts to East Tennessee for a one day conference open to all developers. This is not a trade show with slick salesman giving prepared demos - this is a gathering of real programmers learning about the latest in technology from each other. Sign up now to learn how you can join us for CodeStock 2008!

Speakers and Agenda

We have lined up 30 sessions, a keynote address by respected author and MVP Jeff Prosise, and a place to hang out with speakers and attendees in open spaces style. Check out the list of amazing sessions by awesome speakers.

CodeStock After Party

All work and no play, makes Jack a dull programmer! Hang out with speakers and attendees and enjoy live music by Hanover Fist at the CodeStock After Party.

Registration

Registration is open! You can register for CodeStock 2008 now, and reserve your seat at the greatest developer's conference Knoxville has ever seen!
CodeStock 2008 Developer's Conference

CodeStock's mission is to bring the best and brightest code experts to East Tennessee for a two day event - not a trade show with slick salesman giving prepared demos! - this is a gathering of real programmers learning about the latest tools and techniques from each other. Sign up now to learn how you can join us for CodeStock 2008!

Speakers and Agenda

We have lined up 30 sessions, a keynote address by respected author and MVP Jeff Prosise, and a place to hang out with speakers and attendees in open spaces style. Check out the list of amazing sessions by awesome speakers.

CodeStock After Party

All work and no play, makes Jack a dull programmer! Hang out with speakers and attendees and enjoy live music by Handful of Sugar at the CodeStock After Party.

Registration

Registration is open! You can register for CodeStock 2008 now, and reserve your seat at the greatest developer's conference.
Topics in 2008

WCF
SQL Server
XNA
ASP.NET MVC
AJAX
WPF
LINQ
Open Source
Topics in 2019

- WCF
- REST
- SQL Server
- XNA
- Unity
- ASP.NET MVC
- AJAX
- SPA Apps
- WPF (.NET Core 3)
- LINQ
- Open Source

@JeremyLikness
My Journey to this Keynote
June 2010

Jeremy Likness 🐱@jeremylikness - 25 Jun 2010
Just wrapping up breakfast before I head over to #codestock

Jeremy Likness 🐱@jeremylikness - 21 Apr 2010
Thanks for the support - I'll be speaking at CodeStock this year: http://bit.ly/o0tA0C Looking forward to seeing some of you there!

Jeremy Likness 🐱@jeremylikness - 12 Mar 2010
Voting for CodeStock has begun! Here is my session: http://bit.ly/ko53h Advanced #silverlight applications using #mef

Jeremy Likness 🐱@jeremylikness - 26 Feb 2010
Replying to @gblock
@gblock I submitted a MeF talk for codestock

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
Great time at CodeStock today. Nice to finally meet @wooldruff (Nellie @mecontrol member). Good meeting @DavidGard and others as well.

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
Repeating to @iverkamp
RT @iverkamp It's interesting to see an Entity Framework session from an admitted Ruby hacker. #codestock

Jeremy Likness 🐱@jeremylikness - 25 Jun 2010
@iverkamp Enforcing @oburlton's 'NoSQL I can't relate to my database anymore' session. #codestock +1

Jeremy Likness 🐱@jeremylikness - 25 Jun 2010
Wow very powerful routing and failover features in #Mvc 4.0 #codestock

Jeremy Likness 🐱@jeremylikness - 13 Jul 2010
Received feedback from #CodeStock event. appreciate everyone who took the time to share.

Jeremy Likness 🐱@jeremylikness - 03 Jul 2010
Great explanation of contravariance and covariance by @pkimmel - even if he did prep it at 1AM ! #codestock

Jeremy Likness 🐱@jeremylikness - 28 Jun 2010
Back from CodeStock and it was all about community: http://bit.ly/ATELyo (John Petersen)

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
"Damn! big thumbs up to @nunk, @whbn, and all the folks who made #Codestock 2010 a fantastic event. thanks!" +1

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
Blogged: Advanced #Silverlight applications using the Managed Extensibility Framework AMEF: http://bit.ly/1y01h #codestock

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
Thanks all who were present at my talk. I appreciated you! The deck/source are here: http://bit.ly/jTkDp #codestock #silverlight

Jeremy Likness 🐱@jeremylikness - 26 Jun 2010
Great example of "functional construction" with LINQ converting SQL query to XML by @pkimmel #codestock
June 2010

Jeremy Likness @jeremylikness · 13 Jul 2010
Received feedback from #CodeStock event. Appreciate everyone who took the time to share it.

Jeremy Likness @jeremylikness · 3 Jul 2010
Kicks off her new blog with a #Codestock 2010 recap. Go read and learn the power of Women in Technology.

Jeremy Likness @jeremylikness · 28 Jun 2010
Back from #CodeStock and... http://bit.ly/aT6LYo (John Petersen)

Jeremy Likness @jeremylikness · 26 Jun 2010
"@mamund: Big thumbs up to @viNull, @wbm, and all the folks who made
Personal Timeline ...

2010: Silverlight and MEF

2011: Silverlight 5 for Business

2012: MVVM

2014: Zone.js (Angular)

2016: Angular 2 and TypeScript

2016: Agile DevOps with Visual Studio Online

Visual Studio Team Services

Azure DevOps

@JeremyLikness
The four stages of acceptance...
“Computer Man”
“The Early Years”

¿Cuál es su número de póliza?
“The Early Years”

1994
But what happened in between?
Designing #Silverlight Business Applications #Kindle edition available today amzn.to/GYQnr1 Hardcopy in just a few days.

Jeremy Likness @jeremylikness
4:33am - 29 Mar 2012
Finally Microsoft admits that Silverlight is Dead! Microsoft: The Web is better without plug-ins zite.to/xTOn6e

Llewellyn Falco @LlewellynFalco
10:04pm 31 Jan 2012
Programming the Windows® Runtime by Example
A Comprehensive Guide to WinRT with Examples in C# and XAML

Jeremy Likness
John Garland
"This book doesn’t just focus on singular concepts, it also provides end-to-end perspective on building an app in WinRT. It is one of those essential tools for Windows developers that will help you complete your software goals sooner than without it!"

—Tim Heuer, Principal Program Manager Lead, XAML Platform, Microsoft Corporation
A Cloud Advocate ...

Community
• Meet developers where you are: in person, online, through OSS projects and more

Content
• Create and curate content that empowers you to be successful and achieve more

Connect
• Engage with engineering to close the feedback loop, both to you and from your projects

@JeremyLikness
How do we measure and forecast?

@JeremyLikness
Data
(base)
10 Megabyte Hard Disk

$3,495*

340h-12 Tap Emul Drive

* Factory sealed 10MB cartridge disk drive only

A new Censys Data Systems controller is available for $1,495.

$5,895 for a branded new Amiga 500MB drive only.

We are the CF-94** and AF-94** specialists of Southern California. We can supply you with the latest CF/AM (250) or AF/AM (100) and units Standard B005 ($750) or Duplex B003 ($650). Immediate delivery worldwide. Domestic and foreign importers welcome - delivery time.

*CF/AM and AF/AM are trademarks of National Machine.

We are a full service computer retailer. We totally integrate hardware and software into high-quality, high reliability systems. Systems for use in development, process control and general business. West packaging naturally, fully testing and fully processing too.

COMPUTER COMPONENTS

3400 Bonnievale Boulevard, Van Nuys, California 91401. 213-486-1411
Let’s break that down ...

10 MB = $3,495

My laptop has 1 TB

1 TB / 10 MB = 1000000 MB / 10 MB = 100,000

100,000 x $3,495 = $349,500,000 for storage!
The Query

```
SELECT D.DOMAIN, COUNT(*)
FROM DOMAINS AS D
INNER JOIN HOSTS AS H ON H.DOMAINID = D.ID
INNER JOIN SHORTLINKS AS SL ON SL.HOSTID = H.ID
INNER JOIN CLICK AS C ON C.SHORTLINKID = SL.ID
GROUP BY D.DOMAIN
ORDER BY D.DOMAIN
```
Disconnect: Relational vs. Domain

JeremyLikness
Challenges

1. Schema changes
Challenges

1. Schema changes
2. Complicated joins
Challenges

1. Schema changes
2. Complicated joins
3. High volume inserts (referential integrity)
Challenges

1. Schema changes
2. Complicated joins
3. High volume inserts (referential integrity)
4. Overall cost of the solution
Have you ever ...

• Created a table named metadata and populated it with columns that have names like:
  • Column_name
  • Column_type
  • Column_string_value
  • Column_integer_value

• Stuffed XML, JSON, or other miscellaneous data into a large nvarchar(max)?

• Better yet, used the XML/JSON field types and tried to write a query over them?

@JeremyLikness
Introducing Not Only SQL: NoSQL
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartitionKey</td>
<td>String</td>
<td>e</td>
</tr>
<tr>
<td>RowKey</td>
<td>String</td>
<td>e2y</td>
</tr>
<tr>
<td>Timestamp</td>
<td>DateTime</td>
<td>2019-02-04T18:48:00.337Z</td>
</tr>
<tr>
<td>Medium</td>
<td>String</td>
<td>presentation</td>
</tr>
<tr>
<td>Url</td>
<td>String</td>
<td><a href="http://codestock.org/?WT.mc_id=codest">http://codestock.org/?WT.mc_id=codest</a></td>
</tr>
<tr>
<td>Capacity</td>
<td>= $0.07</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>1 GB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage transactions</td>
<td>= $0.36</td>
<td></td>
</tr>
<tr>
<td>1000 Transaction units (10,000 transactions)</td>
<td>× $0.0004 Per unit</td>
<td></td>
</tr>
<tr>
<td>Sub-total</td>
<td>$0.43</td>
<td></td>
</tr>
</tbody>
</table>
The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. Choose classic deployment model.
// strategy to insert the new short url entry
async Task saveEntryAsync(TableEntity entry)
{
    var operation = TableOperation.Insert(entry);
    var operationResult = await tableOut.ExecuteAsync(operation);
}

// code to retrieve an entry
TableOperation operation =
    TableOperation.Retrieve<ShortUrl>(partitionKey, shortUrl);
var result = await inputTable.ExecuteAsync(operation);
// strategy to insert the new short url entry
async Task saveEntryAsync(TableEntity entry)
{
    var operation = TableOperation.Insert(entry);
    var operationResult = await tableOut.ExecuteAsync(operation);
}

// code to retrieve an entry
TableOperation operation = TableOperation.Retrieve<ShortUrl>(partitionKey, shortUrl);
var result = await inputTable.ExecuteAsync(operation);
{  "id": "7fd95a1b-be2e-4f29-8864-f1d103f55128",  "page": "docs.microsoft.com/azure",  "count": 1,  "timestamp": "2017-10-09T17:46:34",  "presentation": 1,  "_rid": "GzYTAO2ucgACAAAAAA==",  "_self": "dbs/GzYTA==/colls/GzYTAO2ucgA=/docs/GzYTAO2ucgACAAAAAA==/",  "_etag": "0100b23d-0000-0000-0000-59dcd6d50000\"",  "_attachments": "attachments/",  "_ts": 1507645141}
doc = new ExpandoObject();
doc.id = Guid.NewGuid().ToString();
doc.page = page.TrimEnd(normalize);
if (!string.IsNullOrWhiteSpace(parsed.ShortUrl))
{
    doc.shortUrl = parsed.ShortUrl;
}
if (!string.IsNullOrWhiteSpace(campaign))
{
    doc.campaign = campaign;
}
Region Configuration
JLIKME

* This account is geo-redundant.

Throughput (400 - 10,000 RU/s)

400

Estimated spend (USD): $0.032 hourly / $0.77 daily.

@JeremyLikness
Count by Browser

- Pagefreezer crawler
- Other
- Mobile Safari UI/
- Mobile Safari
- LinkedInBot
- Googlebot
- golabot
- Firefox Mobile
- Firefox
- FacebookBot
- Chrome Mobile
- Edge
- Chrome
- AndersPinkBot

Desktop and Mobile
<table>
<thead>
<tr>
<th>azure.microsoft.com</th>
<th>channel9.msdn.com</th>
<th>blogs.msdn.microsoft.com</th>
<th>github.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>dzone.com</td>
<td><a href="http://www.zdnet.com">www.zdnet.com</a></td>
<td>tattoo.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td><a href="http://www.hanselman.com">www.hanselman.com</a></td>
<td>do...</td>
<td></td>
</tr>
<tr>
<td><a href="http://www.smashing.com">www.smashing.com</a></td>
<td>medium.co...</td>
<td>co...</td>
<td></td>
</tr>
<tr>
<td>dev.to</td>
<td><a href="http://www.micro">www.micro</a>...</td>
<td><a href="http://www">www</a>...</td>
<td></td>
</tr>
<tr>
<td></td>
<td><a href="http://www.infoq">www.infoq</a>...</td>
<td>jn...</td>
<td></td>
</tr>
<tr>
<td>vimeo.com</td>
<td><a href="http://www.telerik.com">www.telerik.com</a></td>
<td><a href="http://www">www</a>...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>youtube.com</td>
<td><a href="http://www">www</a>...</td>
<td></td>
</tr>
<tr>
<td>blog.bitsrc.io</td>
<td><a href="http://www.code">www.code</a>...</td>
<td><a href="http://www">www</a>...</td>
<td></td>
</tr>
<tr>
<td><a href="http://www.dotnetrock.com">www.dotnetrock.com</a></td>
<td>blog.revol...</td>
<td><a href="http://www">www</a>...</td>
<td></td>
</tr>
</tbody>
</table>

@JeremyLikness
<table>
<thead>
<tr>
<th>Link</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>docs.microsoft.com/dotnet/standard/serverless-architecture/index</td>
<td>(serverless eBook)</td>
</tr>
<tr>
<td>vimeo.com/303443966</td>
<td>(3 D’s of Modern Web Development Video)</td>
</tr>
<tr>
<td>blog.jeremylikness.com/get-started-with-angular-on-net-core-2-1-part-one-2effcfe8fae9</td>
<td></td>
</tr>
<tr>
<td>dev.to/jeremylikness/dependency-injection-in-javascript-101-2b1e</td>
<td></td>
</tr>
<tr>
<td>github.com/Microsoft/api-guidelines/blob/master/Guidelines.md#710-response-formats</td>
<td></td>
</tr>
<tr>
<td>github.com/funkia/list</td>
<td></td>
</tr>
<tr>
<td>docs.microsoft.com/aspnet/core/client-side/spa/angular?view=aspnetcore-2.1&amp;tabs=visual-studio#server-side-rendering</td>
<td></td>
</tr>
<tr>
<td>tattoocoder.com/dotnet-new-templates-for-serverless-functions</td>
<td></td>
</tr>
<tr>
<td>blog.jeremylikness.com/from-angular-to-blazor-the-health-app-2e36077d641c</td>
<td></td>
</tr>
</tbody>
</table>
Compute $(r)$
THE COMMODORE 64. ONLY $595.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 kilobytes of memory</td>
<td>(I have 250,000 times more now)</td>
</tr>
<tr>
<td>2 MHz speed</td>
<td>... or ~ 1 MHz when you factor in graphics (I now run 2,600 times faster)</td>
</tr>
<tr>
<td>BASIC language built-in to ROM image</td>
<td></td>
</tr>
<tr>
<td>Disk Operating System (LoadStar) built-in</td>
<td></td>
</tr>
<tr>
<td>16 colors</td>
<td>... or so they claimed</td>
</tr>
<tr>
<td>3 voice synthesized sound</td>
<td></td>
</tr>
</tbody>
</table>
I learned performance on a Commodore 64.

8 bits vs. 64 bits
- 256 vs. 18,446,744,073,709,551,615 (quintillions)

64 kilobytes memory vs. 64 gigabytes
- Let 64 KB = distance to moon
- Let 9 billion miles = distance to the edge of the solar system
- 64 gigabytes would get you from Earth to the edge of the solar system and back 13 times

1.023 MHz vs 2.60 GHz
- If one operation on a 2.60 GHz machine takes 1 second, the same operation will take over 40 minutes on a Commodore 64

@JeremyLikness
The Z-Machine: one of the world’s first widely used virtual machines
1976
- Will Crowther writes *Adventure* on a PDP-10 mainframe

1977
- Tim Anderson, Marc Blank, Bruce Daniels, and Dave Lebling finish writing *Zork* using MDL on a DEC PDP-10
- (Briefly renamed to *Dungeon*, the D&D lawyers prompted a quick return back to *Zork*)
1976
• Will Crowther writes *Adventure* on a PDP-10 mainframe

1977
• Tim Anderson, Marc Blank, Bruce Daniels, and Dave Lebling finish writing *Zork* using MDL on a DEC PDP-10
• (Briefly renamed to *Dungeon*, the D&D lawyers prompted a quick return back to *Zork*)

1979
• The Z-machine specification is written on a coffee table in Pittsburgh. It has virtual memory and the ability to save and restore state. It runs *Zork Implementation Language* (ZIL).
1976
- Will Crowther writes *Adventure* on a PDP-10 mainframe

1977
- Tim Anderson, Marc Blank, Bruce Daniels, and Dave Lebling finish writing *Zork* using MDL on a DEC PDP-10
- (Briefly renamed to *Dungeon*, the D&D lawyers prompted a quick return back to *Zork*)

1979
- The Z-machine specification is written on a coffee table in Pittsburgh. It has virtual memory and the ability to save and restore state. It runs *Zork Implementation Language* (ZIL).

1980
- *Zork* is published by Infocom as first part in a 3-part series and ported to PDP-11, TRS-80, Apple II, Commodore 64, CP/M, IBM PC
1976
• Will Crowther writes *Adventure* on a PDP-10 mainframe

1977
• Tim Anderson, Marc Blank, Bruce Daniels, and Dave Lebling finish writing *Zork* using MDL on a DEC PDP-10
• (Briefly renamed to *Dungeon*, the D&D lawyers prompted a quick return back to *Zork*)

1979
• The Z-machine specification is written on a coffee table in Pittsburgh. It has virtual memory and the ability to save and restore state. It runs *Zork Implementation Language* (ZIL).

1980
• *Zork* is published by Infocom as first part in a 3-part series and ported to PDP-11, TRS-80, Apple II, Commodore 64, CP/M, IBM PC

1997
• The Z-machine specification version 1.0 is published
Billing Option

1 year reserved option is not available for your instance selection.

3 year reserved option is not available for your instance selection.

- Pay as you go
- 1 year reserved
- 3 year reserved

Save up to 40% with Windows Server Licenses you already own. Learn more about Azure Hybrid Benefit to save compute costs.
What if I could just write the code that matters ... and not worry about the rest?
What about ... *Serverless*?!
public static async Task<HttpResponseMessage> Go([HttpTrigger(AuthorizationLevel.Anonymous, "get", "post", Route = "Go/{shortUrl}")]HttpRequestMessage req, [Table("urls")]CloudTable inputTable, string shortUrl, [Queue(queueName: "counts")]IAsyncCollector<string> queue, ILogger log)
{
    if (shortUrl.IsNullOrWhiteSpace())
    {
        return req.CreateResponse(HttpStatusCode.BadRequest);
    }

    shortUrl = shortUrl.ToUpper();
    var operation = TableOperation.Retrieve<UrlData>(shortUrl[0].ToString(), shortUrl);
    var result = await inputTable.ExecuteAsync(operation);

    var url = data.Url;
    await queue.AddAsync(data.RowKey);

    var response = req.CreateResponse(HttpStatusCode.Redirect);
    response.Headers.Add("Location", url);
    return response;
}
[FunctionName("ProcessCounts")]
public static async Task ProcessQueue(
    [QueueTrigger("counts")]string shortCode,
    [Table("urls")]CloudTable inputTable,
    ILogger log)
{
    var operation = TableOperation.Retrieve<UrlData>(
        shortCode[0].ToString(), shortCode);
    var result = await inputTable.ExecuteAsync(operation);
    if (result != null && result.Result is UrlData data)
    {
        data.Count++;
        operation = TableOperation.Replace(data);
        await inputTable.ExecuteAsync(operation);
    }
}
What about Scale?
Now for some Twitter integration ...

```javascript
// You need to set your own keys and screen name
var oAuthConsumerKey = "superSecretKey";
var oAuthConsumerSecret = "superSecretSecret";
var oAuthUrl = "https://api.twitter.com/oauth2/token";
var screenname = "aScreenName"; // Do the Authenticate
var authHeaderFormat = "Basic {0}";
var authHeader = string.Format(authHeaderFormat, Convert.ToString64(Encoding.UTF8.GetBytes(Uri.EscapeDataString(oAuthConsumerKey)) + ":" + Uri.EscapeDataString((oAuthConsumerSecret))));
var postBody = "grant_type=client_credentials";
HttpWebRequest authRequest = (HttpWebRequest)WebRequest.Create(oAuthUrl);
authRequest.Headers.Add("Authorization", authHeader);
authRequest.Method = "POST";
WebResponse authResponse = authRequest.GetResponse();
// deserialize into an object
TwitAuthenticateResponse twitAuthResponse;
using (authResponse)
{ using (var reader = new StreamReader(authResponse.GetResponseStream()))
{ JavaScriptSerializer js = new JavaScriptSerializer();
var objectText = reader.ReadToEnd();
twitAuthResponse = JsonDeserialize<
TwitAuthenticateResponse>(objectText);
} } // Do the timeline
var timelineFormat = "https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name={0}&include_rts=1&exclude_replies=1&count=5";
var timelineUrl = string.Format(timelineFormat, screenname);
HttpWebRequest timelineRequest = (HttpWebRequest)WebRequest.Create(timelineUrl);
var timelineHeaderFormat = "{0} {1}";
timelineRequest.Method = "GET";
WebResponse timelineResponse = timelineRequest.GetResponse();
var timelineJson = string.Empty;
using (timelineResponse)
{ using (var reader = new StreamReader(timelineResponse.GetResponseStream()))
{ timelineJson = reader.ReadToEnd();
} }

public class TwitAuthenticateResponse
{ public string token_type { get; set; }
public string access_token { get; set; }
}
```

@JeremyLikness
Is there a better way?
Top Three Tweets (based on clicks)

Jeremy Likness
@jeremylikness

Introducing "List" - a JavaScript fast immutable list (with TypeScript support):
- Immutable
- Faster than arrays
- Chained methods
- Curried functions
- Compatible with tree shaking

Check out the project here: jlik.me/dfz

Jeremy Likness
@jeremylikness

I wrote an eBook about serverless apps architectures.
- Architecture and implementation approaches
- Serverless architecture overview
- @AzureFunctions @logicappsio and Event Grid
- Durable functions
- Business scenarios and use cases

Read it: jlik.me/duv

Jeremy Likness
@jeremylikness

Even if you already have JavaScript skills, you probably think Node.js is too simple. It is. If you're looking for something powerful, you may want to make your site cloud-powered. @MarkHollandCode and @JohnMcCook are capable of turning Node.js into the thing you need.
What About the Front End?
Let’s be honest. I didn’t really even need a framework. HTML5/CSS/JS for the win!
WebAssembly

Your Browser as an OS?

- AssemblyScript
- Blazor (C#)
- C++
- Go
- Python
- Rust

Shameless Plug

@JeremyLikness
Stepping Back...
Serverless Link Shortener

- **ShortenURL**
  - Insert
  - $0.75
- **Redirect**
- **Enqueue**
  - Read
  - $1.40
  - Dequeue
- **Insert**
  - $25.00
- **UpdateTwitter**
  - Read
  - $0.60

@JeremyLikness
What did I learn?

• ☁ It’s possible to build a cloud app on a shoe string budget

• 🫚 Relational databases aren’t the solution for everything

• ✅ It’s OK to have multiple data stores that fit your needs

• 🎁 Today it’s more possible than ever before to focus more on code and less on servers *cough* serverless *cough*

• 🗣 What I think people want to hear is irrelevant. Data is honest.

@JeremyLikness
Technology is growing faster than ever

... but you have a strategy to stay on top of it

@JeremyLikness
“Life is like going up the down escalator. If you stand still, you fall behind.” – yeah, I quoted me
Thank you.

Visit my blog: https://jlik.me/fa8
(of course I had to use a short link)

Serverless: https://jlik.me/fme