

How do you write stuff for a website in a future-proof way? I'm not talking about the HTML and layout. I'm talking about the content. The english language part of it. The lowest common denominator is plain text. HTML itself has pretty good compatibility going back more than 20 years, but not every feature survived. Remember when you could play MIDI files as background music? Or play Flash animations?

Last year in 2023 I redesigned my website. It was a tedious process because I had to double-click the old HTML files, select all the text, copy it, and then paste the text into a shiny new markdown editor.

No Hore they	v•• • • € { }, ss( ), ss ( ), ss ( ), ss ( ), ss ( ), ss	and the bar the transformed and the second s
+ + + + + <u>-</u> - ==	Hand a b P Santration	🖉 👘 🔶 + + + 🖡 📭 🚛 i måld 🛛 =   6 🖉 Suntanää
1 145	ana 🔤 🖬 🚛 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹 🕹	Bertsche ingeste - Music (2005)
	appinent country address,	STATES IN INCOME IN A STATE OF THE STATES AND A STATES
	for anotes a contraction (1995) (1995	Sector Se
I family want larges	galley instruktionieni () editionieni (2003-2004)	and a second secon
30.	p indjenijska 🖉 aktivelja 🗍 👔 🖓 👘	(CAS) ■ search only works with 'spinal covies' ■ rdd content link and
a contract	rendraps 💦 👔 a production 👔 a differente, a 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓 🖓	이 2018년 1월 1월 1월 1월 1월 1월 1월 18일
m	pada pantalan 📕 kajintin 👘 kadawar 🖉 🖓 🖓 🖓 🖓	STATES C. States and S
		No. 2 Context tolde
		A Strain I III Strain Constraint Strain Stra
V parels front		S ( ) ≤ (0 ) =   panla forst - □ X = 0.5( ) ≥ (0 ) = 0.5( ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) ≥ (0 ) = 0.5( ) =
No. Name Ones	🗤 🔹 🔹 and fan Dere Minere and Angeler Der Die er	The first the set of a set of
		PRESS PRESS
+ + + + <mark>-</mark> +-	• 🚥 🔹 • 🚺 (A.A.) (A. 1917) (2017) (A.A.) (A. 1917) (2017)	승규님, 1. (18) 🔹 🔹 📲 💶
plans -		X G - D X
- india.	0 miles x +	File Site Rangingh Format View Themes Hido
apalation.	← → C (0) Fite [Stray(C)/www.twebsite/content/games/game	Hade a pathole particle to service the frame was not the one of project to give you
ida.live		an idea of the insane kind of effort I put into my projects back then, I ripped a bunch of
Chone Ngger_spellas.slp	📕 fie tooring 🧧 YouTube 🚷 Ornal 🚷 Doople Images 🦷 Active website 1 👘 👘	A Drawn Nyger spitter de Marie spitter de Marie spitter for the game using an SNES emulator, molified 28 of her spitters to make
geerle.knotzip	"End" button doesn't do work because I never thought of an ending. You can skip the title-	perfut.xeetre     her half-sude_ and another 73 for when she's fully raked So that's about 93 splice 1     individually allered in Photoshop! Ealso began creating G2F, primations in order to plan the
🔜 guardial const. Sin sig 🖬 migra	response if you click the screen. And the help bulkon will give you hints screetimes.	
Ingo Milipg	This all started as an insurally ambitious project for the "French Kiss" digital paper-doll medium,	<ul> <li>protecting</li> <li>scenes and determine where things would be in each traine. After discovering Macramedia</li> <li>Flash, the GE animations became unrecessary.</li> </ul>
Thurburg	(sha Fixias) Remember these flat dress-up paper della? Fixias was is digital version of it that got	
B guardia forestand	advanced enough to support crude assistation and interaction. I get into it since people saturally made a branch of underesting games with it. I originally plasmed for this game to be made in	The game used to use midi files for music, but I can't really de that on most websites. Plash
and appleters and	Fixes Then I descentered Maccounting Flash. Making this in Flash made a lot more sense. It	games earmally have to be self-contained. So 1 changed it to use MPG's. 1 didn't create the restanced restance from DCreative.
	seemed like Flash was made for this kind of project! To give you as idea of the instate kind of	music. Most of it came than Dohanix.
	effort I put into my projects back then. I report a bunch of Marle speaks from the game using an	ostatitas o
	ness condutor, modified 23 of her sprites to make her half ende and another 73 for when she's fully solved? So that's about 93 sprites I individually altered in Photosboy? I also began creating.	
	The average of the second of spinor of an even and determine where things would be in each for	Amoldan Amoldan Amoldan Amoldan Amoldan
	After discovering Macromodia Flash, the OIF minutions because unseconsary	file-"http://corweis.org/veris/UCR81851/" origin-"Unipulator"
	The game used to use mide files for music. But I can't early do that on most weld. A Flash	orlginLink="http://screetx.org/net/s0088863/"sc/mad.chrcs
		And a construction of the second of the second se
	rassie: Most of it came from OCremin. Copy	Card Card Table Card Table Card , da/0081178/" origin="barksSore"
Circles 1 Train second 2.	Copy lots in highlight	Copy ( Paul Paul Cory Projector)/00/01/30/1/30/most dires
	Insuch Daub/Daub/In for "This is a could blir her	the game based on Devenue." I see a state of the second seco
	<ul> <li>Jano Zeal Love by Unipulator</li> <li>Fest.</li> </ul>	Chief Chief and Chief Ch
	Entro Zoal Love by Unipulator     Forest     Forest Through the Dark by DarkeSound	
125	<ul> <li>Jano Zeal Love by Unipulator</li> <li>Fest.</li> </ul>	

It's not the first time. 14 years ago in 2013, I also redesigned my website. I made a custom editor using Flash to generate JSON for my JavaScript single-page-app website. But that website wasn't all JSON. Each project also had a readme file. A simple HTML file with minimal markup. So I double-clicked each of the older HTML files, selected part of the text,

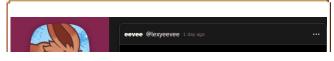
copied... and then pasted it into that project's HTML readme file. This seemed to make sense. HTML is basically just XML right? Surely I could just use an XML program to read the data in the future.



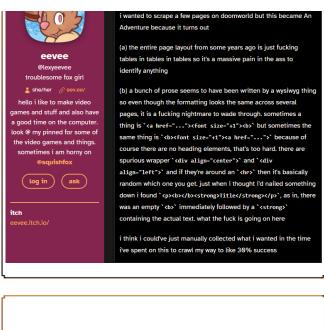
It turns out that HTML is *not* proper XML. Maybe XHTML was going to be, but that didn't really take off. To put things in perspective:

- In 1997 HTML was the format of the future.
- In 2005 XML was the format of the future.
- In 2014 JSON was the format of the future.
- In 2016 Markdown was the format of the future.

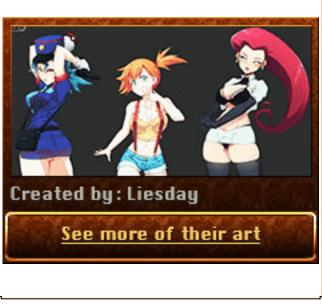




Why didn't I just use the HTML I already had? One problem was that my old website was inconsistent. In spite of mu efforts to use



minimal HTML, it was still all hand-written and so the same things would get written in slightly different ways on different pages. A computer program would struggle to recognize them all, and I would struggle to remember all the places I wrote something.



ile Edit Paragraph Format View Themes Help

I also had new ideas when I redesigned my website in 2023, about what *kind* of information I wanted to store, and how I wanted to present it. I often recommend artists. How should I store a "recommendation?" It's a very niche and specific concept, so there is no single dedicated HTML tag for this. And being an abstract concept, it can also be presented in multiple ways. All equally valid. Should it be a semantic FIGURE tag with a fig-caption? But what if I want to include a link with it?

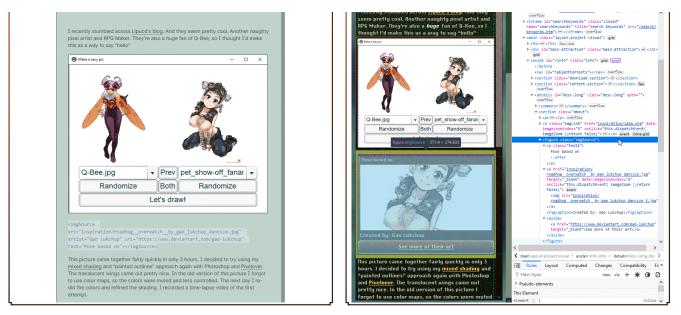
I also learned more about semantic HTML, <u>schema microformats</u>, and new CSS features that make <u>new layout methods</u> possible. What I needed was a more generic way to represent things, and a separate way to handle their presentation. It's always a good idea to separate content from presentation, so that you can change how things look without having to modify ALL the content. That's why CSS was invented in the first place, but it's not enough by itself. And even though HTML lets you add tags with arbitrary names, I still need to include things like normal HTML links inside of them. The real problem is that composing *multiple tags* together runs the risk of creating consistency. I can easily write them in any order. But what if I could represent something with a single tag instead?

So here I am in 2023 / 2024 storing my website's source as Markdown, and using <u>HUGO</u> to convert it into HTML. I now represent abstract things like artist recommendations using single "XML" tags because I found a way to make Hugo interpret these as short-codes so that it replaces them with whatever set of tags I want when it generates the HTML files. And if I lose that option in the future, any other markdown editor will harmlessly passthrough these "tags" into the HTML, where something else could potentially handle them.

About this project

🖂 🗇 … ×

🕞 🖸 Inspector ▷ Console ≫



This has two advantages. HUGO will always write this HTML exactly the same way every time, and my markdown files store the content in a form that is even simpler than HTML, without combining tags together.

Hugo isn't limited to generating HTML either. It can write any kind of text file. So in another 10 years when I (very likely) redesign my website again, I *assume* I can probably just use HUGO to convert the markdown into the next text format "of the future"... assuming computers can still run windows programs written in x86 code, but <u>that's not guaranteed</u>. I suppose there's always emulation.





Part of my inspiration came from reading about how Cory Doctorow handles his blog. He is a very prolific blogger who has used <u>XML</u> for over 20 years and uses various python scripts to convert his articles for his website and social media posts. That probably works,



but I don't envy having to edit XML markup by hand.