

ViennaDoc:

An Animatable and Testable Specification Documentation Tool

Tomohiro Oda

Hiroshi Sako

Keijiro Araki

Han-Myung Chang

Yasuhiro Yamamoto

Peter Gorm Larsen

Kumiyo Nakakoji

outline

- background
- demo
- ViennaDoc: overview
- html
- discussions
- summary

background

Who cares functional specifications?

- formal specification engineers
- architects/designers/implementors
- test engineers
- document (manuals, references,...) writers
- UI designers
- development managers
- product owners
- end user representatives

ViennaTalk is an environment for exploratory specification

- **communication media**
 - with UI designers ... Lively Walk-Through
 - with web frontend engineers ... Webly Walk-Through
 - **to fill gaps** between formal engineers and **stakeholders with less formal engineering background**
- powered **by animation**
 - using VDMJ
 - using Smalltalk transpiler
 - provides web API for animation

demo

ViennaDoc: overview

formal specification and specification document
should be **engaged**

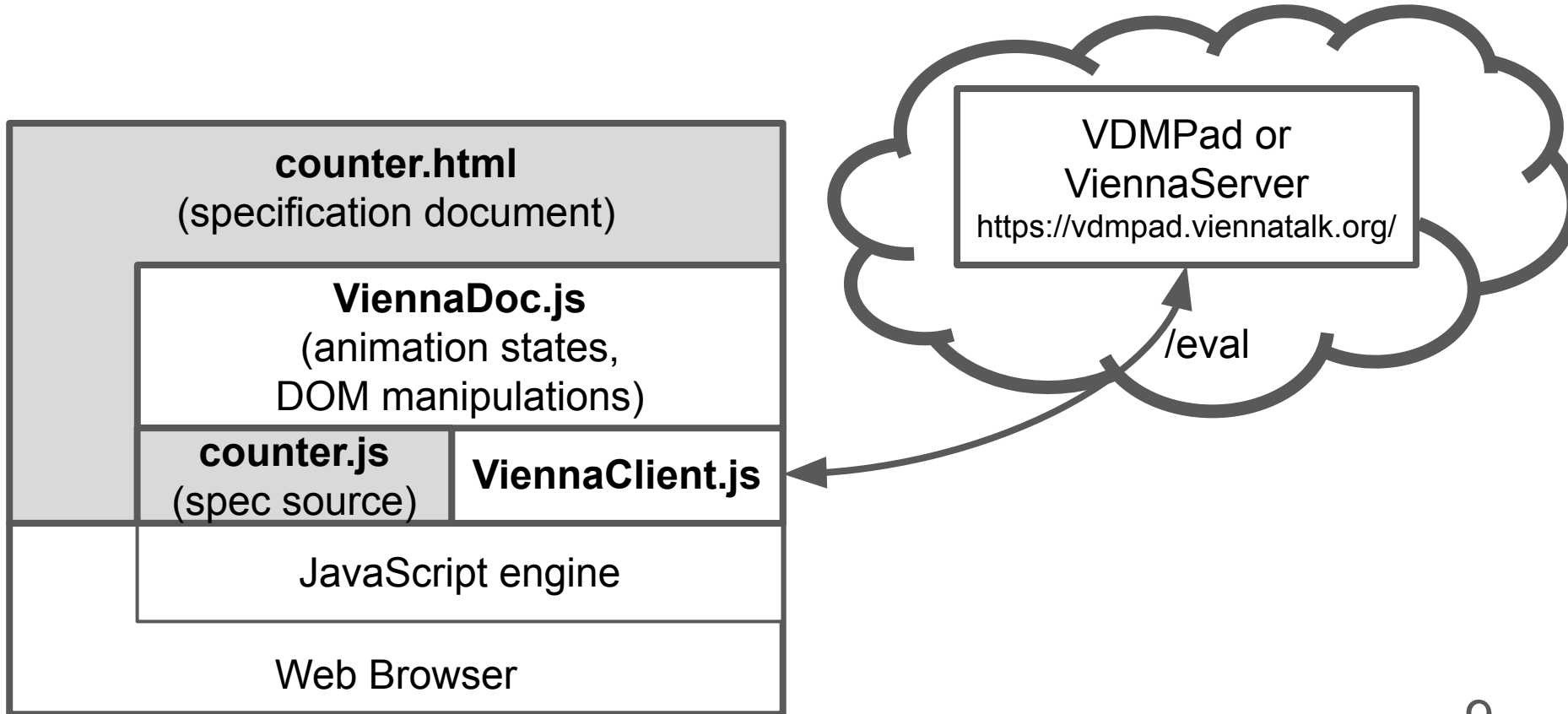


operations

```
inc : () ==> nat -- increments  
inc() == (count := count + 1;  
return count);
```

The `inc` operation
increments the count and
returns it.

ViennaDoc = HTML + VDM animation



ViennaDoc

JavaScript library enables HTML documents

- to include specification source,
- to insert buttons to evaluate quoted expressions, and
- to test assertions on the browser.

ViennaDoc

JavaScript library enables HTML documents

- to include specification source,
→ to keep the document up-to-date
- to insert buttons to evaluate quoted expressions, and
- to test assertions on the browser.

ViennaDoc

JavaScript library enables HTML documents

- to include specification source,
- to insert buttons to evaluate quoted expressions, and
→ to explain by animating the formal spec
- to test assertions on the browser.

ViennaDoc

JavaScript library enables HTML documents

- to include specification source,
- to insert buttons to evaluate quoted expressions, and
- to test assertions on the browser.
→ to confirm consistency with the formal spec

html

load

```
<script src="ViennaClient.js"></script>  
<script src="ViennaDoc.js"></script>  
<script src="Counter.js"></script>
```

source inclusion

```
<code vienna="source" src="Counter`inc">  
</code>
```


evaluation

```
<code vienna="eval"  
      prestates="count"  
      poststates="count"  
      module="Counter">  
  
inc()  
</code>
```

watch expression

```
<code vienna="watch" module="Counter">  
counter  
</code>
```

assertion

```
<code vienna="assert"  
    module="Counter"  
    prestates="count=6"  
    eval="mk_(reset(), count).#2">  
0  
</code>
```

discussions

ViennaDoc

vs

LaTeX literate styles

- VDM spec and html managed sperately
- best for online browsing
- processed on a browser (testing/animating)

- one can choose either monolithic or modular
- best for printing
- processed at build-time (e.g. testing a doc)

LaTeX / ViennaDoc / VDMPad

	Overture w/ LaTeX	ViennaTalk w/ ViennaDoc	VDMPad
read VDM source	○	○	○
read informal text.	○	○	△
eval a given exp	○/x	○	○
eval a free exp	○/x	○/x	○
test a document	x	○	x
test a spec	○/x	○/x	△
large spec	○	○	x
VDM++/VDM-RT	○	x	x

Specification document in the "programming-as-common-literacy" era

Hypothetical assumption:

- In near future, **most stakeholders will have** a certain level of **programming skills**.
- We don't have to **hide** formal specifications. We can simply **explain** them in **a plain specification language** and **a plain natural language**.

The background of the slide features two large, interlocking gears. The gears are rendered in a dark gray color with a subtle gradient, giving them a three-dimensional appearance. They are positioned symmetrically on either side of the central text.

Summary

Formal Specifications and Specification Documents

Summary

ViennaDoc provides specification documents

- to **include VDM source** for the constructs of concern,
- to **exemplify behaviors** of a particular functionality, and
- to **validate the document** by assertions

for **readers with programming literacy skills**

demo page:

<https://viennatalk.org/ViennaDoc/counter.html>

We need **more!**

- diagram notation like VDMPad
- previews on ViennaTalk
- in-browser authoring
- testing infrastructure on CI
- support for handy notations, e.g. markdown, pillar
- export to static documents, e.g. epub, LaTeX, pdf
- integration / collaboration with Paul's HTML generator
- VDM-SL language tutorial