

# Digital examinations of the future using HTML5

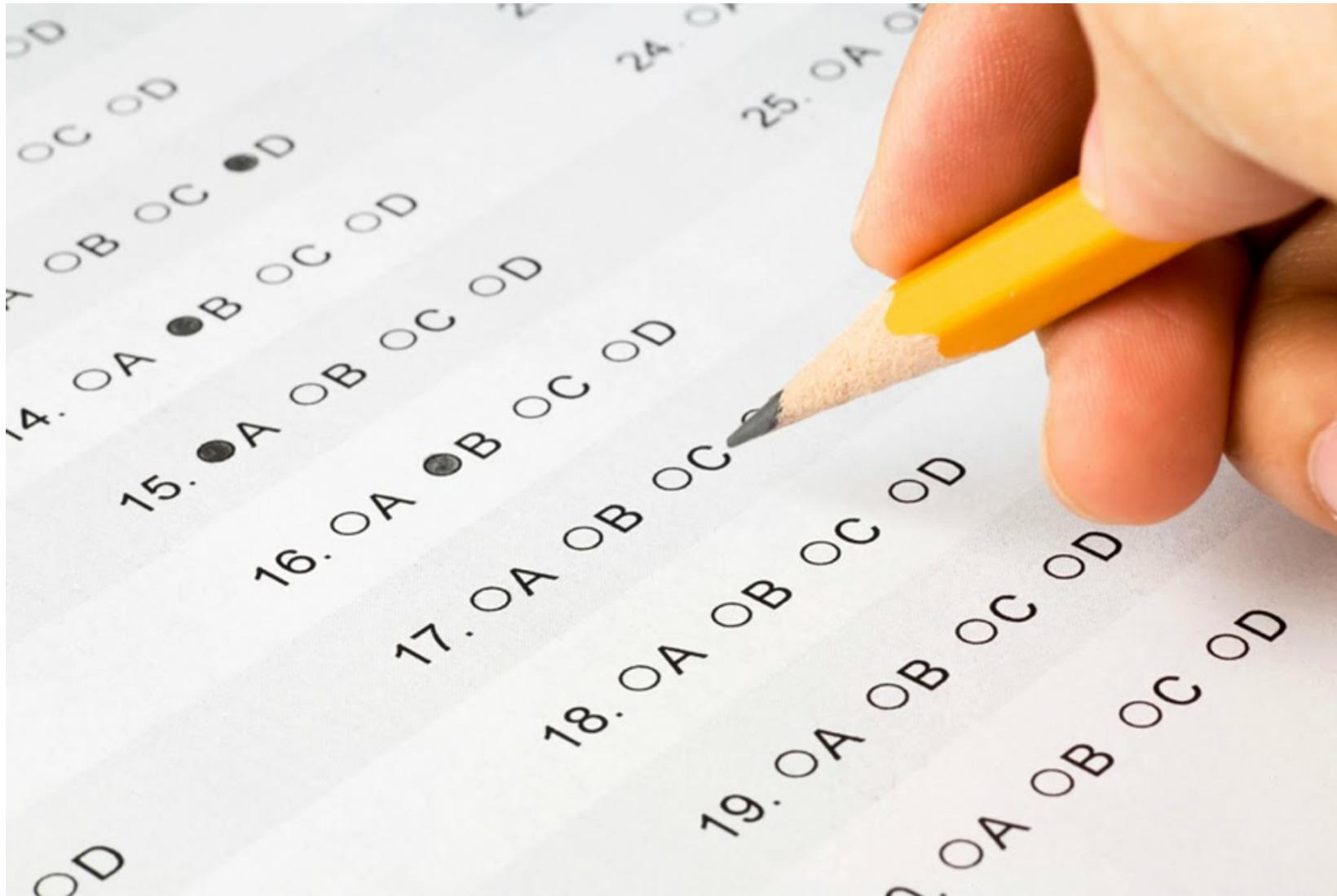
---

**Sjoerd Mulder, GOTO Amsterdam 24-04-2012**

Front-end engineer at Orange11

# Digital examinations?

---



# Central Exam System (CES) Project

---

- College voor Examens
  - Cito: content delivery
  - DUO: infrastructure, storage, logistics
  - Orange11: end-user interface
- Multi-year project

Successor of the ExamenTester from Cito
- Special requirements

availability, security, cross-platform, open-source

# History: Cito ExamenTester

---

- In use since 2004
- Windows desktop only



**foto 1**

Het oplezen van de Troonrede



Op de foto's zie je koningin Beatrix twee taken verrichten.

**Op welke foto zie je een taak van de koningin als hoofd van de regering?**

- alleen bij foto 1
- alleen bij foto 2
- bij foto 1 en 2
- bij geen van beide foto's

**foto 2**

Het openstellen van de Oosterschelde dam



## Requirements / features

---

- High concurrent system, 30.000+ concurrent users
- No server sessions
- Lock-down client

## Next-generation application

---

- Greenfield
- 2014+
- Open standards
- Consistent cross-platform interface

## Which technology would you choose?

---

- Adobe Flex / Flash
- Microsoft Silverlight
- HTML5

# Technology stack

---

## Server:

- Java, Java Servlet 3.0, StAX
- Spring MVC (REST), Freemarker

## Client:

- HTML5, SVG, CSS3, Javascript
- jQuery, jQuery UI
- RequireJS, Spine (MVC), Mustache

*Major part of rendering logic is done client-side*

# IMS Question and Test Interoperability (QTI)

- XML standard for the representation of test content and results
- IMS Global Learning Consortium

```
<orderInteraction responseIdentifier="RESPONSE" shuffle="true">
    <prompt>
        The following F1 drivers finished on the podium in the first ever
        Grand
        Prix of Bahrain. Can you rearrange them into the correct finishin
        g order?
    </prompt>
    <simpleChoice identifier="A">Rubens Barrichello</simpleChoice>
    <simpleChoice identifier="B">Jenson Button</simpleChoice>
    <simpleChoice identifier="C" fixed="true">Michael Schumacher</simpleC
        hoice>
</orderInteraction>
```

```
<responseDeclaration identifier="RESPONSE" cardinality="ordered"
    baseType="identifier">
    <correctResponse>
        <value>C</value>
        <value>A</value>
        <value>B</value>
    </correctResponse>
</responseDeclaration>
```

# Screenshot (orderInteraction)

The following F1 drivers finished on the podium in the first ever Grand Prix of Bahrain. Can you rearrange them into the correct finishing order?

Jenson Button

Rubens Barrichello

Michael Schumacher

```
<responseVariable xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"  
identifier="RESPONSE" cardinality="ordered" baseType="identifier"  
choiceSequence="DriverB DriverA DriverC"><candidateResponse>  
<value>DriverB</value><value>DriverA</value><value>DriverC</value>  
</candidateResponse></responseVariable>
```

## Live example (choiceInteraction)

---

**Which of the following elements are used to form water?**

- Carbon
- Helium
- Hydrogen
- Oxygen
- Chlorine
- Nitrogen

```
<responseVariable xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"  
identifier="RESPONSE" cardinality="multiple" baseType="identifier"  
choiceSequence="C He H O Cl N"><candidateResponse/></responseVariable>
```

Let's do a short exam...

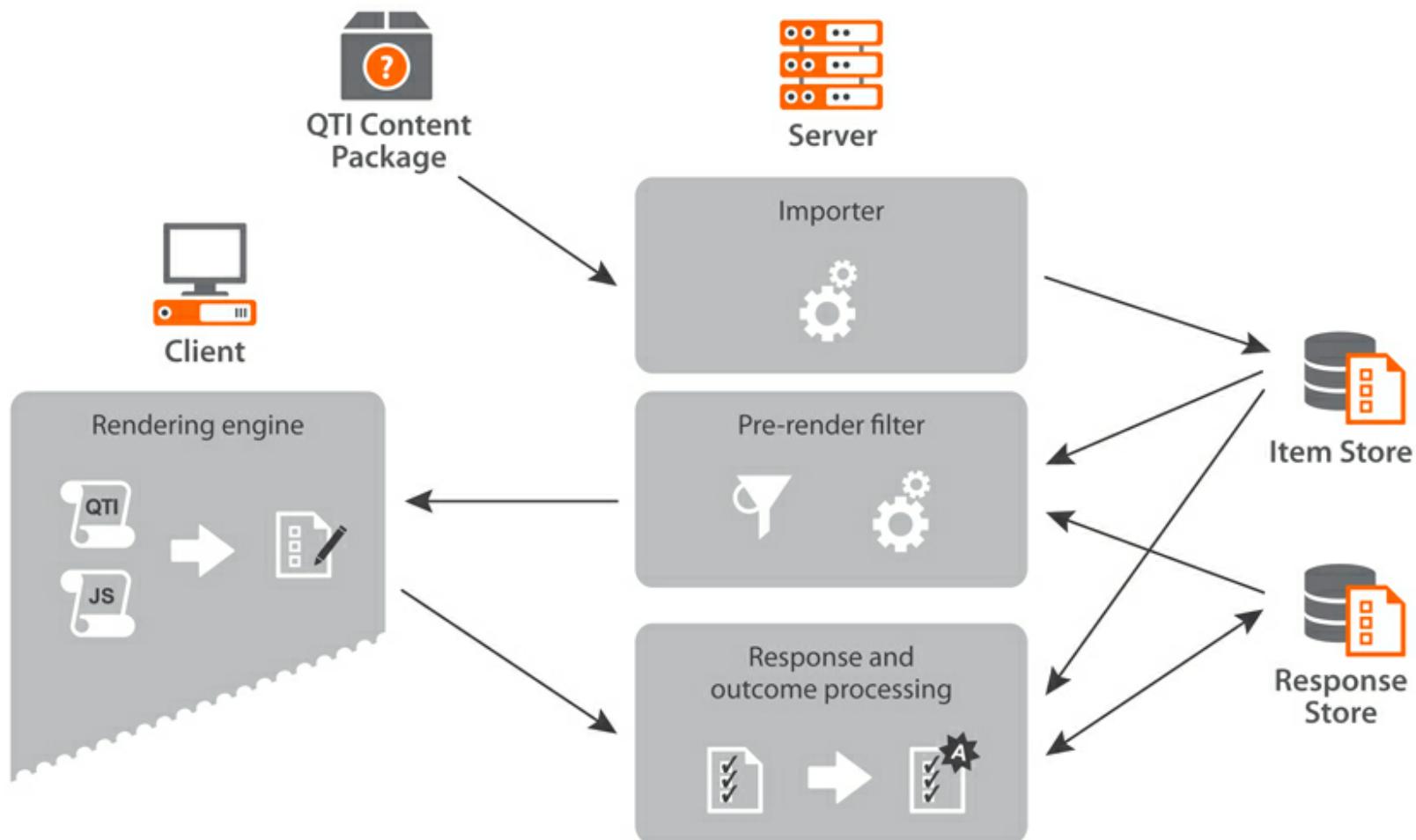
---

# How did we build it?

---

# Architecture overview

---



# Rendering Engine

---

- It's more than just a script
- Dependency Management
- MVC

Minimize inter-dependencies

- HTML = markup, CSS = styling, Javascript = logic
- A class shouldn't do private stuff with an other class

Don't

```
$('<div style="border:1px solid #9f0;">Hi</div>').appendTo('.someClass');
```

# RequireJS: dependency management

```
//DepA.js
define(function callback() {
    var privateProperty = "Some value never exposed";

    var MyClass = function(name) {
        //constructor
        alert("Hello " + name);
    };
    return MyClass;
}) ;
```

```
//main.js
require(["DepA"], function callback(DepA) {
    //The following line will alert "Hello world"
    var item = new DepA('world');
}) ;
```

```
//DepC.js
define(["DepA", "DepB"], function callback(DepA, DepB) {
    //Do something with DepA and DepB
    return DepA.extend(DepB);
}) ;
```



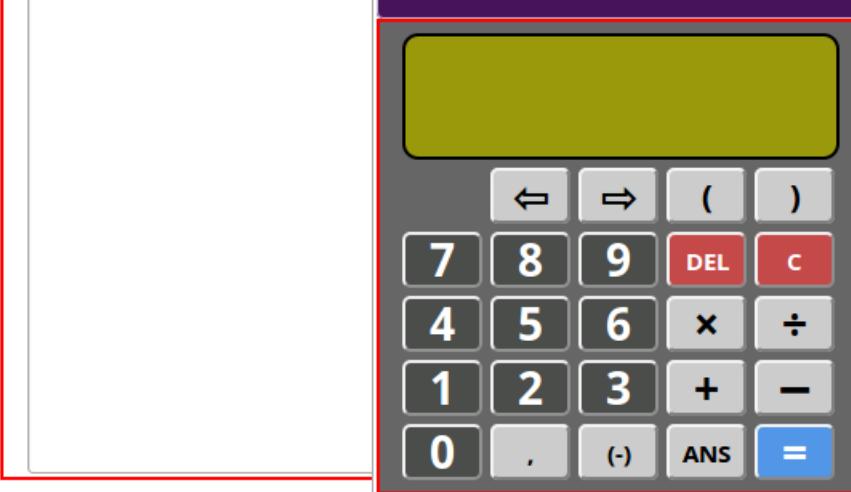
Which of the following elements are used to form water?

- Carbon
- Hydrogen
- Helium
- Chlorine
- Oxygen
- Nitrogen

Kladblok



▼ Rekenmachine



Examen Engels VMBO (kb-en-vb-10)



Which of the following

- Carbon
- Hydrogen
- Helium
- Chlorine

&lt; Vorige

1

2

3

4

5

6

7

8

9

10

11

12

Overzicht

Volgende &gt;

# Spine: model view controller pattern

```
define(['ces-data', 'spine', 'model/notepad'],
  function (data, Spine, NotepadModel) {

  var Notepad = Spine.Controller.sub({
    elements: {
      "textarea": "textarea"
    },
    events: {
      "keyup": "handleChange",
      "change": "handleChange"
    },
    init: function() {
      NotepadModel.bind('refresh', this.update);
      NotepadModel.fetch();
    },
    update: function() {
      if (NotepadModel.exists(data.vraagId)) {
        var model = NotepadModel.find(data.vraagId);
        this.textarea.val(model.contents);
      }
    },
    handleChange: function(event) {
      var model = NotepadModel.getExistingOrNew(data.vraagId);
      model.contents = this.textarea.val();
      model.save();
    }
  });
  return Notepad;
}) ;
```

# Notepad model

---

```
define(['spine', 'model/persistable'], function(Spine, Persistable) {

    var NotepadModel = Spine.Model.sub({ }, {
        /**
         * Gets an existing or new record
         */
        getExistingOrNew: function(id) {
            if (NotepadModel.exists(id)) {
                return NotepadModel.find(id)
            }
            return new NotepadModel({id: id});
        }
    );
    NotepadModel.configure("Notepad", "contents");
    NotepadModel.extend(Persistable);
    return NotepadModel;
}) ;
```

# QTI Delivery Engine

```
<orderInteraction responseIdentifier="RESPONSE" shuffle="true">
    <prompt>
        The following F1 drivers finished on the podium in the first ever
        Grand
        Prix of Bahrain. Can you rearrange them into the correct finishin
        g order?
    </prompt>
    <simpleChoice identifier="A">Rubens Barrichello</simpleChoice>
    <simpleChoice identifier="B">Jenson Button</simpleChoice>
    <simpleChoice identifier="C" fixed="true">Michael Schumacher</simpleC
        hoice>
</orderInteraction>
```

```
define(['imsqti_v2p1/abstract/bodyElement'], function(BodyElement) {
    /**
     * The prompt element is stating the actual question, it will be
     * transformed into a HTML P tag
     * @name imsqti_v2p1.Prompt
     * @extends imsqti_v2p1.abstract.BodyElement
     */
    var Prompt = BodyElement.sub/** @lends imsqti_v2p1.Prompt.prototype
*/ {
        tag: 'p'
    };

    return Prompt;
});
```

## Summary

---

- You can develop desktop-like apps in a browser
- Don't stop using standards and best practices
- An end-user interface is "that thing" people see

Treat it with respect, take its code serious!

---

The browser is a great development platform, maybe  
the only platform for applications in the future?

---

Want more information about QTI & the delivery engine?



- Visit our stand to view the demo,
- contact us at [qti@orange11.nl](mailto:qti@orange11.nl),
- or visit our website [www.orange11.nl](http://www.orange11.nl)

