Dec3D: Declarative 3D for the Web Architecture

Kristian Sons
DFKI
Saarbrücken, Germany

Johannes Behr
Fraunhofer IGD
Darmstadt, Germany

Philipp Slusallek
DFKI
Saarbrücken, Germany

ABSTRACT

With the advent of WebGL and Adobe Stage 3D, 3D capabilities have finally arrived in the major Internet browsers. These imperative solutions are tied to the functionality of rasterization APIs, thus the usage require a deeper understanding of the rasterization pipeline. In contrast to this stands a declarative approach with an abstract description of the 3D scene. Such an approach fits perfectly into the HTML5 and related technologies: The DOM can be used for the scene structure, CSS for style etc. Web developers can easily apply their knowledge to 3D content and thus a much broader usage becomes possible. This paper characterizes the need for such an approach, sketches its functionality and points out connections to related technologies from W3C and other standard bodies.The position presented was developed during the starting phase of the W3C Community Group (CG) *Declarative 3D for the Web Architecture*.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

Keywords

3D Graphics, World Wide Web, Declarative Approach

1. INTRODUCTION

It is arguable that the emerging support for an imperative 3D-API for the Web is useful but insufficient for broad acceptance and usage of 3D on the Web. A declarative approach that is tightly integrated with current web technologies and that offers qualified concepts is necessary to support a fast adoption and broad use of interactive 3D graphics by the millions of existing Web developers. The provided concepts must lift the hardware-oriented imperative application programming interfaces (APIs, e.g. OpenGL/WebGL [5]) to an expressive and more easily usable level. Therefore not the low-level data structures of existing hardware layers must be in the center of the design but high-level elements and items like 3D objects, transformations, material descriptions, and lights. Instead of teaching Web developers 3D graphics APIs, the goal is to bring 3D graphics to the point where its natural for Web developers to just make use of it. While this might not be possible for every possible use of a low-level API, we believe that it can cover the vast majority of use cases.

Powerful graphics hardware is already in every PC and most mobile devices, stereo displays are heading from the big screen to the consumer space, and Internet connection speeds are generally fast enough also for 3D data. What is still missing is an easy way to add interactive high-level 3D objects to HTML to allow anyone with a Web browser to easily create, share, and experience interactive and possibly immersive 3D graphics in any Web browser – with possibly wide ranging effects similar to those caused by the broad availability of video on the Web.

The objective of the Declarative 3D for the Web Community Group is to explore and suggest options for integrating a declarative approach to interactive 3D graphics directly into HTML and thus enable its use in any Web page. This work will also consider how declarative 3D graphics can fit into the larger XML and Web architectures in a manner comparable to Scalable Vector Graphics (SVG, [**?**]) or Mathematical Markup Language (MathML, [**?**]).

2. DECLARATIVE VS. IMPERATIVE 3D

ToDo: I believe that comparing Declarative and Imperative approaches is very important part of this paper, as WebGL is now a really hot topic (Jacek).

2.1 Declarative Approach

Declarative 3D is being developed to significantly lower the barrier for authoring 3D content for Web sites. It aims to lower the barrier by duplicating the key features that enabled the growth of the early Web:

No installation, configuration, maintenance - Anyone could "join the Web" simply by putting an HTML file on a web server. Similarly with Declarative 3D, we envision people only need to put an HTML file on a standard web server and their audience immediately experiences that data through a rich browsing user interface.

Separation of structure from content - Underlying the Web from its earliest days was the separation of structure from content. The concept of a paragraph specified by the <p> tag was separate from the content in the paragraph. Declarative 3D is attempting to bring the same separation of structure from content to 3D graphics inside of web pages.

Separation of content from style - One of the principles of the current Web is also the separation of content and style, most notably through CSS. The successful integration of SVG with HTML was made much easier due to the fact that SVG was already following this principle.

A "simple as possible" user interface - 3D navigation techniques, even simple ones, can be very difficult to understand in the context of the Internet hypertext environment... We need standard UIs.

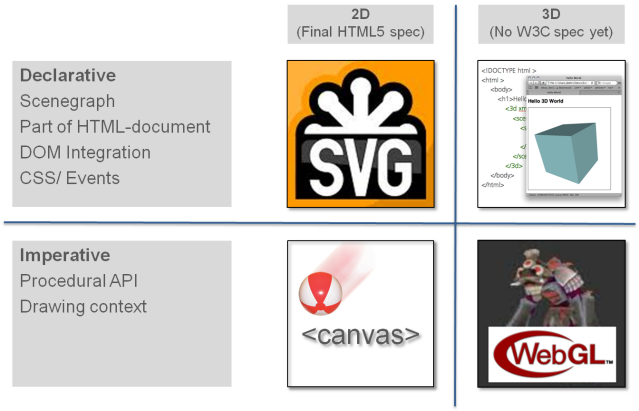

Figure 1: Declarative vs. Imperative.

2.2 Imperative Approach

Buuu...;)

3. PREVIOUS ATTEMPTS: VRML & X3D

ToDo: In my opinion we need to be clear about the roots of Declarative 3D. We need to clearly state where we position ourselves (Jacek).

The idea of integrating 3D computer graphics declaratively into the Web infrastructure has already been discussed. Back in the 1990s there was a huge R&D activity devoted to the Virtual Reality Markup/Modelling Language (VRML), a simple, text-based format for describing virtual worlds. This term was coined by Dave Raggett in a paper submitted to the First International Conference on the World Wide Web in 1994, and first discussed at this conference at VRML BoF that was organized by Raggett and Tim Berners-Lee[1]. Shortly afterwards, in November 1994, the first version of VRML was specified.

To further manage the development of the format, the VRML Consortium[2] has been formed. In 1997, a second version of the format was finalized, as VRML97 (also known as VRML 2.0) and has been accepted as an ISO international standard. VRML entered its third generation as X3D [2], which adds XML capabilities to better integrate with other World Wide Web technologies.

This early web standard provoked much interest but has never seen much serious widespread use. We believe that there were few reasons for this:

Plug-in Dependency - VRML needed a viewer (plug-in) to run in a Web browser - users had to deal with plug-in installation (security and compatibility issues);

Lack of Bandwidth - Slow (in that time usually dial-up) internet access resulted in users having to wait for long periods of time (sometimes even minutes!) to download VRML scenes;

Poor Graphics Performance - Low frame rate did not provide the illusion of smooth motion - animations did not look realistic; this affected the user's willingness to interact with virtual environments;

Low Usability - Exploring VRML environments was (and still is!) difficult, especially from the point of view of the new-to-3D users. Inadequate support to user navigation often resulted in users getting *"lost-in-space"* and ultimately leaving the site.

Poor Integration with a Web Architecture - VRML was all about 3D graphics; it did not *"fit in"* into a DOM-based architecture and HTML-based Web interface.

In recent years, some of these problems seem to have been solved. Today, 3D acceleration is a standard feature on most video cards. Bandwidth is not an issue anymore (at least in most *"developed"* countries...).

Some problems are being solved: WebGL, a successor of Canvas3D [5], is a JavaScript API specification analogous to the OpenGL API for C++ [3]. It aims to bring plugin-free 3D to the web, implemented right into the browser; at the time of this writing, WebGL is supported in most major web browsers: Apple's Safari, Google's Chrome, and Mozilla's Firefox.

Finally, some issues still need to be considered and addressed. Declarative 3D is a solution to the *"Poor Integration with a Web Architecture"* problem.

4. NEW ATTEMPT: DEC3D

ToDo...

4.1 The Community Group

Integrating interactive 3D into the Web includes aspects such as

- Integration with existing and future W3C standards, specifically HTML5 [**?**],
- Integration of high-level scene concepts,

[1]Extending WWW to support Virtual Reality: http://www.w3.org/People/Raggett/vrml/vrml.html
[2]Web3D Consortium: http://www.web3d.org/

- Integration into user agents,
- Integration with suitable interaction techniques and devices,
- Integration of 3D content on the server side,
- Integration with client and server side rendering services,
- Integration with existing security aspects for content delivery and visualization

The Community Group aims at creating the necessary technical and organizational prerequisites to eventually start a Working Group for a recommendation-track specification at the W3C. Towards that end, hosting a workshop to collect detailed information about requirements, use cases, metrics and potential exemplary applications can provide great value. This approach has worked well for other W3C efforts including Video on the Web and XML Binary Compression (XBC)/Efficient XML Interchange (EXI) [**?**]. Workshop participants often become the principal contributors to follow-on Working Groups. In the long term, Declarative 3D Graphics appears to be a good fit for the W3C Graphics Activity.

4.2 Scope

The goal of this Community Group (CG) is to evaluate the options for a successful standardization of a declarative approach to interactive 3D graphics as part of HTML documents. The goal is to collect suitable use cases, derive requirements from them, and then find the essential set of features and concepts that enables broad uptake by authors and users of interactive 3D on the Web. This group is aimed at the rapid development of a lean, extensible, and well-understood set of features necessary to cover the majority of useful 3D use cases for the Web – but not necessarily all of them. Therefore, one of the key tasks for the XG will be evaluating and rating those use cases to develop suitable requirements. If requirements are well defined, then a survey of candidate technologies can demonstrate whether a follow-on Working Group is likely to achieve the defined goals. Based on these requirements, features and necessary new will be proposed and classified with respect to their importance for the use cases. If successful, a staged approach for their possible standardization will finally be developed. Any selected feature or concept should be orthogonal to current Web functionality and to each other wherever possible. They should be based on and leverage existing W3C technologies (i.e. CSS, DOM, DOM Scripting, and DOM Events) wherever applicable. For its integration into HTML and for possible future extensions, the group should make use an extension mechanism that might be provided by a future version of HTML (or develop its own if necessary). The XG for now does not explicitly target the high-end game applications or others use cases that have the knowledge and the resources to cope with the typical low-level API-based solutions and typically want full control of the graphics engine. It should rather focus on the many other existing and novel applications that seek ways to create, publish, and share interactive 3D assets and to build 3D applications based on the W3C technology stack.

4.3 Goals and principles

The following goals and principles should guide the development:

Definition of meaningful and future-proof scene concepts

Define expressive concepts that abstract from graphics APIs and provide a high-level declarative scene description to Web application developers and support the selected use cases and requirements. This could include elements like objects, transformations, surfaces, material abstractions, and lighting.

Extension to HTML

The concepts should be implemented in a declarative 3D description as an extension to HTML5 using any existing or future extension mechanism. It should closely follow the established principles of Web technology and W3C standards.

Leveraging Existing W3C Standards

The XG should reuse existing W3C techniques (specifically from HTML5 and SVG) as far as possible and will propose extensions only where 3D specific features are necessary or where they provide significant benefits. Where new concepts are introduced their relation to and effects on existing Web standards should be analyzed, evaluated, and discussed with the respective W3C working groups.

Content Portability and Platform Independence

The aim of the XG is to describe 3D content in a way that does rely neither on a specific render API such as OpenGL or DirectX nor on a specific rendering technique such as rasterization or ray tracing only. The technology investigated should allow for content to be portable across user agents, rendering techniques, and hardware platforms, while taking advantage of available features wherever possible. The results of rendering content under such different environments should be highly predictable.

3D Content Creation

While the creation of original 3D geometry and appearances still requires 3D specific know-how, the reuse, configuration, and manipulation of such content should be made similarly easy as for 2D Web content now. The solution should hide internal data structures and algorithms and provide users convenient ways to edit and manipulate such scenes.

Tool Chain, Other 3D File Formats, and Semantics

A key success factor for Declarative 3D on the Web will be the ability to generate new or reuse existing content. This requires that suitable exporters and converters can be built. However, as 3D on the Web is supposed mainly as a delivery mechanism, it is not necessary to include the ability to semantically represent all but the important 3D features

Efficiency and Scalability

Interactive 3D graphics operates in real-time, which enables new forms of interactivity on the Web but also adds significant new requirements on user agents. A key requirement for the selected technology therefore is the possibility to implement it efficiently. Since 3D scenes can sometimes

become large, any solution should target scalability in the sense that 3D content should run across different platforms (from mobile devices to high-end graphics hardware) with predictable performance. Mechanisms should be in place to handle cases where the performance provided by a user agent on some platform is not sufficient, e.g. by allowing for switching to different content (e.g. LOD) or provide alternate methods of delivering the content (e.g. server-based rendering delivered via streaming video).Security and

Digital Rights Management

Secure delivery of Web content is a general problem and not specific to 3D data. However, the economic value of 3D data might make the problem more acute. Any proposed solution should therefore be based on a general approach to secure Web content. The XG will, however, collect use cases, extract requirements and examine how far existing methods and standards can be transferred to the proposed architecture. It is already demonstrated that application of XML Encryption and Signature is needed for document fragments as well as full documents, since high-fidelity or sensitive portions of 3D models often need special protections.

Accessibility

A large body of work has shown that accessibility improvements serve all users, not just people with disabilities. A curious aspect of many 3D graphics approaches is that user navigation is implemented inconsistently. Therefore, users familiar with one approach are impeded when navigating or interacting with other 3D scenes and models. Examination of relevant Web Accessibility Initiatives (WAI) principles might provide significant benefit. Conversely, use of declarative 3D graphics models might provide major benefits when describing the accessibility features and constraints of real-world objects and locations. Declarative 3D goals and potential solutions may achieve significant benefits if they are harmonizable with WAI imperatives.

4.4 Success Criteria

The XG will be successful if it can

- agree on a collection of use cases and their classification, dependency, and priority,
- derive and prioritize a set of requirements from them, identify the possible new concepts and features necessary to cover the requirements,
- propose possible solutions for each identified feature, define properties that quantitatively or qualitatively measure the quality of the solution,
- document the pros and cons of each solution based on these measurements,
- demonstrate that, based on the above analysis, there is a good chance of success in creating a W3C standard for *Declarative 3D for the Web*.

4.5 Out of Scope

The following topics are not considered part of the charter of this XP:

- Pure procedural API-based approaches. However, providing APIs for convenient usage and connections to pure API solutions (such as Typed Arrays[**?**] and WebGL) is within the scope.
- Solutions not compatible with W3C Recommendations and related Web or internet standards.
- Recommendation-track specifications are out of scope for this Community Group, though it may produce early drafts of specifications and prototypes that later can lead to recommendation-track specifications within a W3C Working Group.

4.6 Deliverables

As its main result the XG will deliver a report documenting its progress, any conclusions it arrived at with respect to standardization of *Declarative 3D for the Web* and, if reaching a positive conclusion, recommending a standardization approach as a basis for a future W3C working group on the same topic. One or more reports that survey and explore possibly different options for adding interactive 3D graphics to HTML, including which use cases and requirements are covered, A list of use cases, requirements, features and concepts, and measurable properties as the basis for its report, Demonstrators for the use cases to show feasibility and exemplar capabilities This Community Group may also produce early drafts of specifications, which may be evaluated as potential recommendation-track deliverables by a chartered Working Group. Dependencies and Liaisons

4.7 W3C Groups

HTML Working Group: Most important dependency, as the tight integration into HTML is a main focus point. Cascading Style Sheets (CSS) Working Group: As leveraging CSS is a key discussion point of the XG, the demonstrators might have to extend existing CSS functionality. Scalable Vector Graphics (SVG) Working Group: The XG coordinates closely with the SVG WG on common features and concepts of 2D and 3D graphics. Web Applications (WebApps) Working Group: DOM Events, XBL, and other publications of the WebApps WG could affect the work in this group and vice versa. Device APIs and Policy Working Group: Especially the integration of camera and other sensor data into interactive 3D scenes is interesting for a wide range of 3D applications, particularly user interfaces and augmented reality (AR). Web Accessibility Initiative (WAI): The Web Accessibility Initiative (WAI) develops strategies, guidelines, and resources to help make the Web accessible to people with disabilities. WAI products also provide interface improvements that benefit all users. Points of Interest (POI) Working Group: The work of this group is strongly related to topics of the POI WG, notably augmented reality, location-based services, and navigation systems.

4.8 Liaisons with External Groups

The following is a preliminary list of external bodies the Working Group should collaborate with: Web3D Consortium: The Web3D Consortium develops X3D, an ISO format for declarative 3D graphics. Khronos Group: The Khronos Group develops 3D graphics APIs such as OpenGL and WebGL as well as the 3D graphic format COLLADA. International Organization of Standards (ISO) standards committees for 3D graphics

4.9 Participation

All relevant stakeholders, e.g. developers, designers, 3D artists, industry professionals, accessibility experts, and user-agent implementers, are encouraged to participate in this Community Group. Participants must be willing to actively develop and donate materials towards the group?s deliverables, as well as attend the majority of the group's teleconferences and face-to-face meetings. We expect that a large number of W3C members will be interested in this effort, either to participate directly or to spur the achievement of long-term goals for declarative 3D graphics on the Web.

4.10 Demonstrator platforms

The XG will start its discussion, demonstrate, and evaluate use cases on the fully working demonstrator platforms and implementations developed by the DFKI (XML3D [4]) and Fraunhofer IGD (X3DOM[1]). Other open demonstrator platforms or other input to the evaluation of *Declarative 3D for the Web* are highly welcome.

5. FUTURE DIRECTIONS

Core research and development objectives for the next years for the Declarative 3D include: (1) the creation of collaboration infrastructures, which will support collaborative work opening up new opportunities for individuals and organizations; (2) Strategy for Validation; and (3) Outreach...

5.1 Collaboration

The current infrastructure provided for the W3C Declarative 3D for the Web Architecture Community Group includes: mailing list, wiki, chat, issue tracking, and RSS channel.

ToDo: We need to learn how to efficiently use what we have. In my opinion we need more support for collaborative work (and more collaborative work..;)). We need to learn from other W3C groups how to work, what tools to use, etc. We need strong leadership with clear agenda. We also need money - we should look for funding in FP7/FP8 (Fraunhofer, DFKI, DERI, ...) and US...

5.2 Constant Validation

The techniques for integrating interactive 3D with HTML cannot be created in a vacuum, but inside a highly dynamic information infrastructure - the Web, which provides us with a living laboratory enabling us to validate our approaches and to improve our ideas.

ToDo: The first way to validate our approaches is to study the usage of emerging WebGL sites..? and investigating the approaches in application domains, ranging from eHealth to eLearning.

5.3 Outreach

ToDo: We need to outreach to HTML, Cascading Style Sheets (CSS), and Scalable Vector Graphics (SVG) W3C Working Groups. We need to contact some Browser Vendors (e.g., through the W3C groups).

6. CONCLUSIONS

In this paper we described what is Declarative 3D, we characterized the need for such an approach, sketched its functionality and pointed out connections to related technologies from W3C and other standard bodies... ToDo: Extend.

ToDo: Honestly discuss risks involved in following our approach. From *"We Tried This Before, and it Didn't Work"* to a loooong way of standardization - min 5-10 years?

7. ACKNOWLEDGEMENTS

Contributors to this document include Johannes Behr, Don Brutzman, Leonard Daly, Anita Havele, Nicholas Polys, and Yvonne Jung. XXX Mitschreiber raus...

8. REFERENCES

[1] Johannes Behr, Peter Eschler, Yvonne Jung, and Michael Zöllner. X3dom: a dom-based html5/x3d integration model. In *Web3D'09*. ACM, 2009.

[2] Don Brutzman and Leonard Daly. *X3D: extensible 3D graphics for web authors*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007.

[3] Khronos. Webgl. http://www.khronos.org/webgl/, 2010.

[4] Kristian Sons, Felix Klein, Dmitri Rubinstein, Sergiy Byelozyorov, and Philipp Slusallek. Xml3d: interactive 3d graphics for the web. In *Web3D'10*. ACM, 2010.

[5] Vladimir Vukicevic. Canvas 3d: Gl power, webstyle (retrived 11.2010). http://blog.vlad1.com/2007/11/26/canvas-3d-gl-power-web-style/, 2007.