



DyVisual
Dynamic Visualization and Interaction

User Guide

Author(s): Klaus Fischer
Copyright: © 2015 DFKI GmbH
Release Date: 1-9-2015
Revision: 0.9



Table of Contents

INTRODUCTION 4

1. OVERVIEW 4

2. ARCHITECTURE AND SPECIFICATIONS 4

3. GETTING STARTED 4

4. USER INTERFACE 5

5. APPLICATION PROGRAMMING INTERFACE 7



Introduction

This document provides a brief description of the user interfaces for the prototype applications which were implemented in the context of the FITMAN project.

1. Overview

DyVisual was applied to two use case applications in the context of the FITMAN project. The implemented solutions are generic in the sense that they can be deployed in similar applications. Additional to the specific solutions for the use cases a generic player for captured motions represented in BVH¹ files.

2. Architecture and Specifications

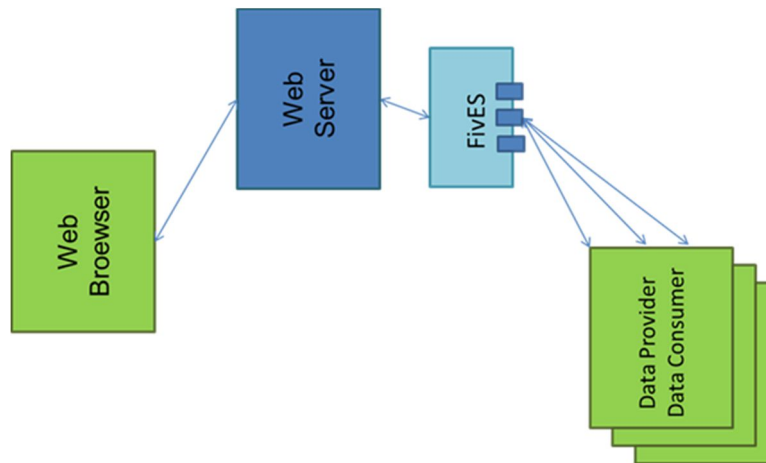


Figure 1: DyVisual Architecture

DyVisual is specialized on the dynamic visualization of 3D models and content. It also provides standard interaction mechanisms. It is for example possible to turn the 3D model zoom in and out and even fly inside the model. Figure 1 displays the architecture of DyVisual. The core components of DyVisual are a Web server and the FiVES synchronization server. Together they allow access to the 3D models for multiple clients. In the first place these clients are Web browsers which render the html pages and 3D models provided by FiVES. Additional clients can directly connect to FiVES and update the information in the server which can result in changes in the 3D scene which are then displayed by the browser clients.

If a user wants to implement his or her own application with DyVisual, he or she needs to dig into how to represent 3D content in XML3D and how to integrate it with FiVES. In this document we concentrate on describing the user's point of view for the TRW and Whirlpool use cases. However, the source code of these prototype implementations is available from SourceForge (see Section 3) and can be used for inspiration for the development of similar or completely new applications.

3. Getting Started

¹ https://en.wikipedia.org/wiki/Biovision_Hierarchy

Please visit <http://sourceforge.net/projects/dyvisual/> and use the “read me” files for the two prototypical use cases for the Whirlpool and TRW use cases.

If you are only interested in looking at the result you can go for the Whirlpool use case to <http://xml3d.org/xml3d/scenes/magnifi/> where you find the standalone Web app for the Whirlpool case. The user interface of this app is described in

4. User Interface

4.1. TRW

In the TRW use case DyVisual was used to visualize results of ergonomic evaluation with the help of a virtual avatar. Depending on the ergonomic evaluation the avatar is put into a specific position and problematic joints are marked with spheres of different colours where red marks that there is high risk and green says that there is no risk.

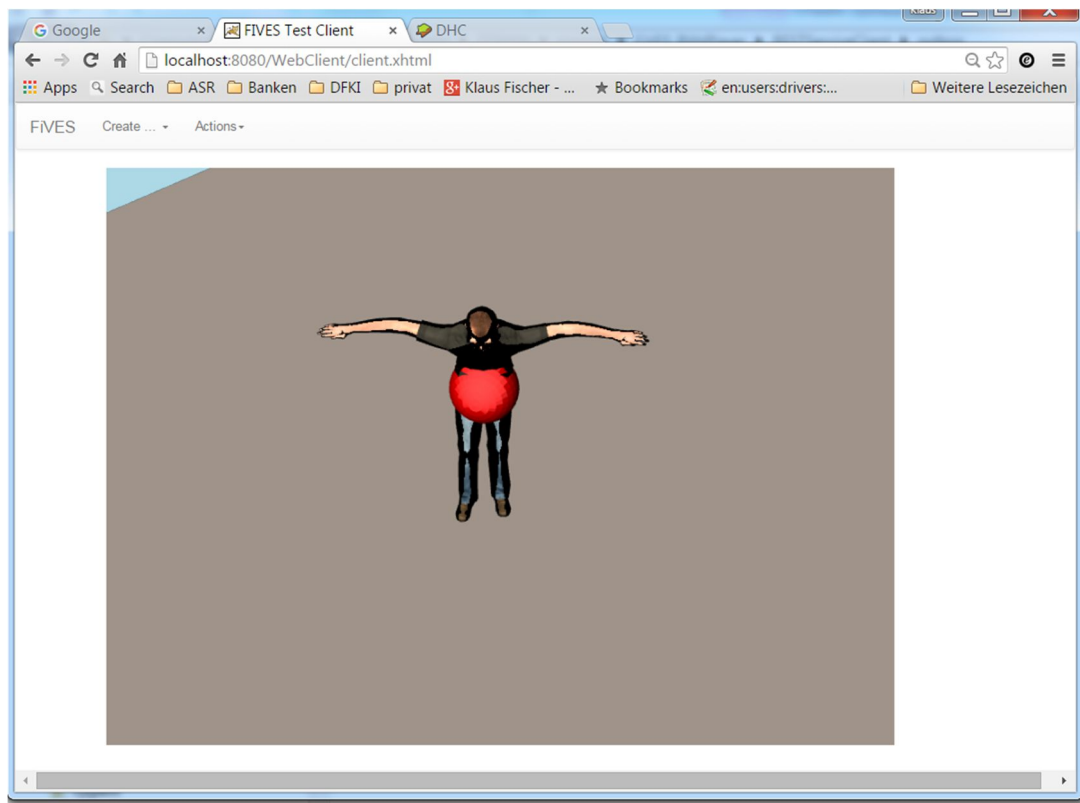


Figure 2: Browser window of the TRW prototype of DyVisual

Figure 1 shows the browser window which offers the interface to the user. With the help of the menus the user can create additional avatars (Create) and add markers on joints of the avatar (Actions->Add Marker) or modify the pose (Actions->Update Pose).

Unfortunately, there is a problem with the prototype implementation. License limitations prohibit that the mesh information of avatar are made public. Problem is that with the mesh information we would give away the model of the avatar where the license agreement only allows us to share this model with partners with whom we collaborate in the context of a project. In case you want to have the model to be able to run the prototype in your

environment, you need to send us email and we can see whether we find an arrangement which would put us into the position to share the model.

4.2. Whirlpool

For the Whirlpool use case there is a standalone Web app which can be deployed in any Web server and basically shows at least some aspects of the visualization features of DyVisual. The fastest way to access the Web app is to go to <http://xml3d.org/xml3d/scenes/magnifi/>.

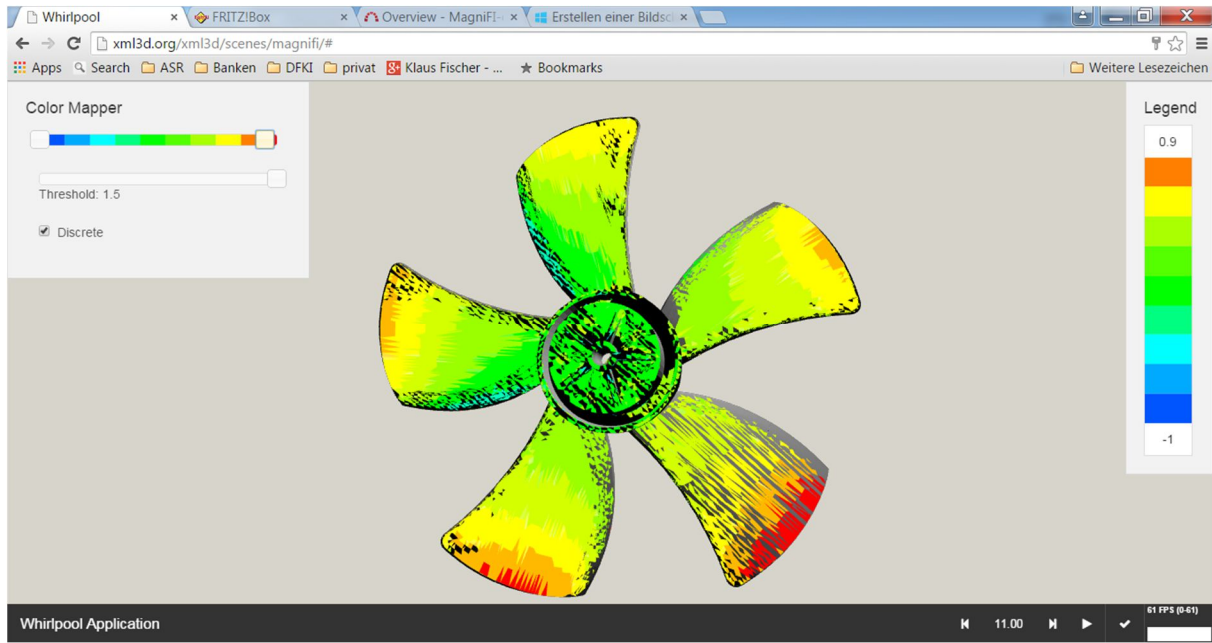


Figure 3: Web App for the Visualization of Deviation Maps

Figure 3 displays a screen shot from this Web app for displaying deviation maps. The idea of a deviation map is that it visualizes the differences (i.e. deviations) of a physical part from the CAD model which was defined for its design. For this, usually a point cloud is produced for the physical part. In the context of FITMAN this point cloud is produced by a high accuracy laser scanner. The point cloud is then compared with the CAD model which results in the deviation map. The scanning and computation of the deviation is not part of the Web app. The Web app purely displays the deviation maps which were already computed.

With the sliders in the top left it is possible to change the setting for the color mapping of the deviation map. The app can visualize a set of deviation maps. With this it is possible to visualize a time line of deviation maps. This can be helpful if a trend, e.g. that there is an increase in deviations of a specific piece of a given part, in the production of the parts needs to be investigated. With the buttons on the lower-right it is possible to switch between the deviation maps. Additionally the user can turn the model by pressing the left mouse button and dragging it over the model. By pressing the right mouse button and dragging it up or down the user can zoom the model in or out, respectively.

The source code for the Web app can be downloaded from the directory WhirlpoolWebApp from the SourceForge pages (see Section 3). The *read me* file explains the installation procedure.

Additionally to the Web app a full version of a DyVisual Web client is provided in the DyVisual in folder WhirlpoolWebClient. The *read me* file explains how the Web client can be deployed. Not all of the features of the standalone Web app are implemented yet. For now the model visualized and can be investigated in the same manner as in the Web app. The next

step is to implement the synchronization of the camera which is the main feature which FiVES provides to this prototype implementation.

5. Application Programming Interface

The prototype implementations for the TRW and the Whirlpool use case both offer restful service interfaces. The signatures of the rest interfaces are described in pdf documents provided with the prototype implementations.

If users are interested in designing their own applications, they need to take a closer look at the documentation of XML3D and FiVES. Together with the code for the prototype implementations for it should be not too difficult to implement complex applications with interactive and dynamic 3D content.

