

## **Webb Based Simulation Service (WBS Service) for motion simulator, a tool study to create Web Based Simulation Portals (WBS Portals) for Virtual Laboratories used in e learning.**

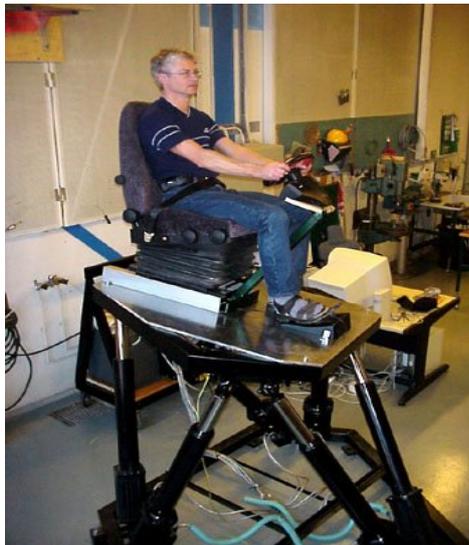
### **Introduction:**

This paper will present an implementation of WBS Services to pre- and post process motion information from a real or synthetic vehicle. The vehicle is reproduced as a 6 degree of freedom motion simulator. Our approach is that the development of the motion simulator, WBS Portal, will allow tools and knowledge to easily be transferred to WBS Portals for distance education and e learning.

We would like to use the definition of Web Based Simulation Service (WBS Service) as an interactive web interface with server hosted scripts. Those will perform calculations based on input from the user and present the results to the user. The definition of a WBS Portal acts as a collection of one or more WBS Services or tools which support an autenticated user with workspace to store notes and results from the calculations. A WBS Portal can also support collaboration and knowledge exchange with other identified users.

Web Supported Simulation Services [ref B2B]), WBS Service [ref WBS] and WBS Portals can be connected to an account system. The Purdue University uses WebCT Vista as a WBS-Portal [ref purdue].

**Background:** In Luleå university of technology we had a Maxcue 6 Degrees of Freedom motion platform, that is usually controlled by a program directly connected to the platform called sweetly “The Shaker”. As we where renewing the hole system, we got an idea “why not share this resource and put it over the net to perform some calculation in a server accessible from a web client” and that was the point for our



first part of the project.  
Simulator with driver.

Fig 1. The Motion

The main idea was to create a website as interface for the world to send all kind of signals to the platform and let a MATLAB server, with MATLAB web toolbox, do some calculation of the signals and send them to the platform controller. It's also possible, but not yet implemented, to do post-process calculation like FFT of feedback

and response signals from the platform also using a web interface interacting to the MATLAB server.

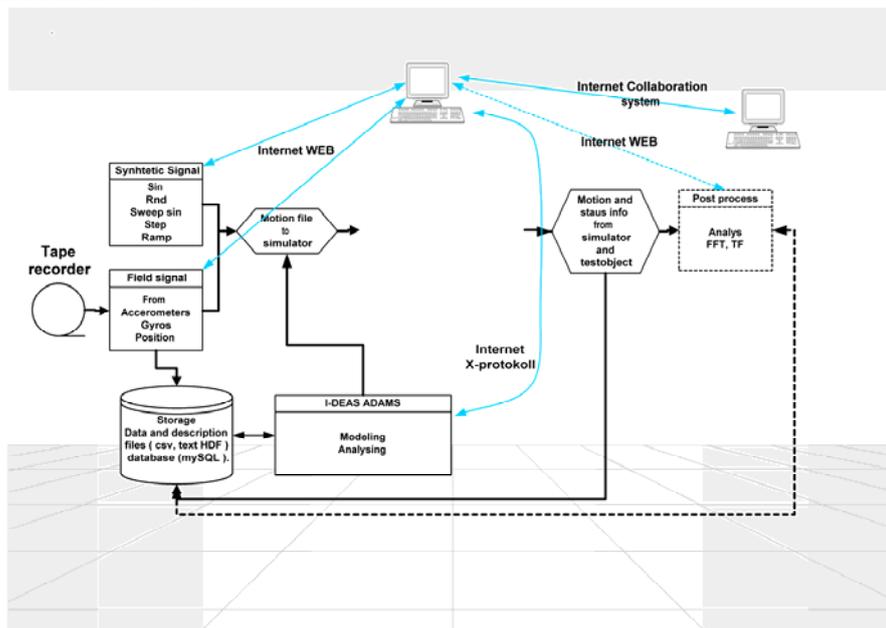


Fig 2. Schematic view of calculation modules, storage and web interaction.

### Approach:

There are two **WBS Services** which prepare the data controlling the motion simulator. One generates the synthetic signal and the other converts the field signal into a motion signal. The motion simulator could be described as the moving base below a flight crew-training simulator. We are using our motion simulator to reproduce a vehicle motion, mostly off road. The goal is to improve the drivers comfort by introducing an active suspended cabin.

In our recently renewed motion simulator, (the previous simulator was constructed in -79,) we decided to make it possible to test and analyse the result from any office at any location [ref SVIB]. This approach gave us the advantage of no need to occupy the physical test plant except when performing the real test. There would be no need for the project staff to be on site, the project manger would be able to present the test and the result for the project owner in any office or boardroom and - finally this technique could be applied as an educational tool.

Our approach on designing the WBS Service had the following elements:

- **Platform independent:** meaning that the code should be able to run on any operating system using any web server. Our choice was PHP [ref PHP], Java and MATLAB [ref MATLAB]. We also did use free source code from Zend PHP Code Gallery [ref ZEND] if present and the code has high quality. This approach is also described in Overcoming Obstacle to e-Learning [ref pierce].
- **Multi Language support:** the user interface should be generated in the users preferred language, using preferred language variable sent from the browser. The server side script should generate the html code in the user's native language that he has defined as preferred language in his web-browser, if present on the server. Presently only English, hope to add Swedish, German and Spanish at the presentation.
- **Distributed services:** if a WBS Service is using separate web scripts and calculation script engines, the calculation script engine are able to run on a

separate server. In this way a web server can use a pool of calculation servers and many web servers may be used to run the same application scripts on one calculation server. A specific calculation script can be copied to many calculation servers and also for example be called from web servers according to a random selection. A calculation server can reside on the same location as the team, which is constructing and supporting the script and customizing questions about the calculation. Any web server can connect to the calculation engine performing the script when authenticated. This approach will also give a high redundancy.

- **Recording of test:** the input and output from the tests will automatically be stored in Hierarchical Data Format [ref HDF] for documentation and export of test data to other programs. The HDF format could be substituted to text file or db storage if the documenting is a result from a calculation performed by for example students.

**Setup:** To make this work, we used a computer with 512MB memory. The software was Windows 2000 Server with Windows Terminal Server (WTS) activated. We inactivated the IIS web server preferring Apache. From MATLAB 6.5 [ref MATLAB] we used the MATLAB web server toolbox according to the setup instruction and all available toolboxes. The terminal server made it possible for the developer to simultaneously logon to the server through WTS clients from anywhere in the world in order to develop new or edit present WBS-Services or WBS-portals. The development programs used on the WTS were EditPlus [ref EditPlus] and MS Frontpage. This was our setup, but there is no need to use a WTS, you could also edit your scripts with your favourite editor on your own workstation and either connect to the simulation server through a share- or a ftp-server or develop your script on the simulation server itself.

To estimate cost and resources needed for this development we used the COCOMO 2000 [ref. COCOMO] method. We substituted Perl (Single Line Of Code SLOC) for PHP and C++ (SLOC) for MATLAB to estimate the coding effort.

Our first step was to create a WBS-Portal with separate frames for: Info about motion simulator, WBS-Services for simulated signals and recorded signals, documentations, useful links, web cam covering the motion simulator and one about the authors [Ref WBS Portal].

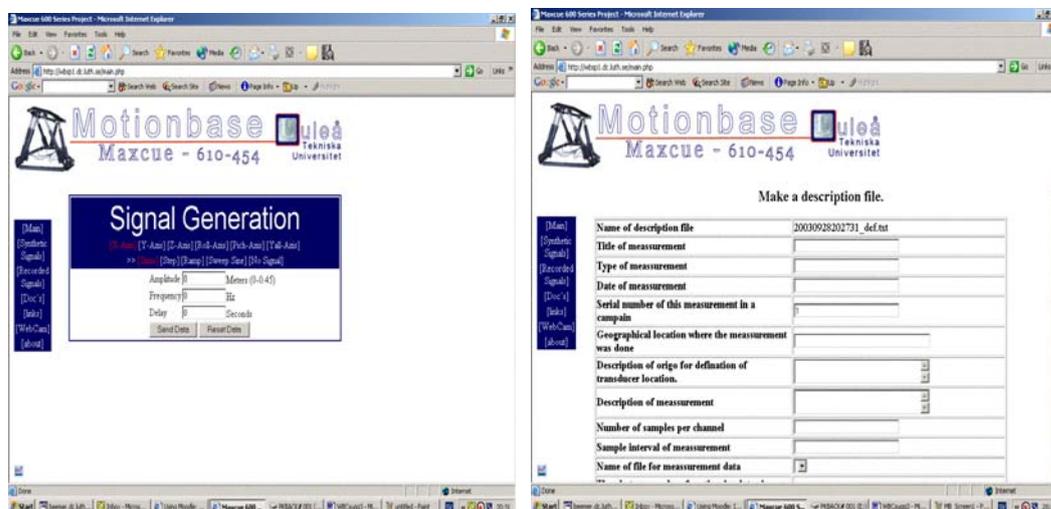


Fig3 Web interface for Synthetic Signal Fig4 Web interface for Field signal.

The synthetic signal WBS-Service uses JavaScript in the client to alter the parameter input form according to different signals (sin, sweep sin, ramp,.) without reloading the webpage. The form transfers parameters to a MATLAB script, as described in the supplied examples, that calculates time domain signals to control the motion simulator. The time domain signal are formatted as a CSV (Comma Separated Values) file accessible as a link on the response web page. The challenge in this WBS-Service, as in the field signal WBS-Service, was to minimize the number of steps in for-loops and in number of loops. Our trick was to create a portion of the signal and repeat that portion over and over again and in some cases (sweep sine) mirror the portion of the signal before adding it to the time domain signal. Mathworks has significantly improved the performance of for-loops in newer versions. There are two main factors that influenced the creation of the field signal WBS-Service, the need to include documentation of recording setup with the recorded data - and reuse of code from previous real-time systems written in Fortran.

The WBS-Service entry web-page allows the user to either select or edit a previous defined field signal **recording** or to define a new field signal **recording**. A definition consists of a label of the recording, where, when, notes and records of the transducers as type, location on the vehicle and amplifications. The definition also contains records at what location relative to the transducers that the motion of an object should be simulated and also the name and location of the recorded time signals.

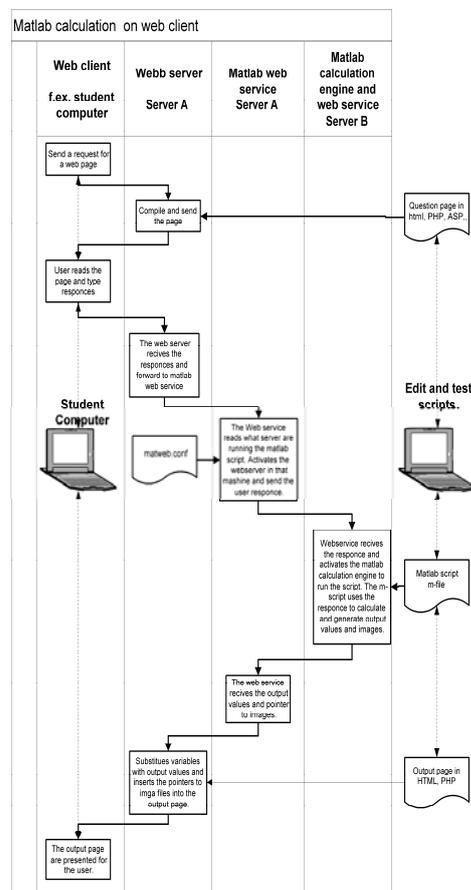


Fig 5 workflow of interacting and editing MATLAB web scripts.

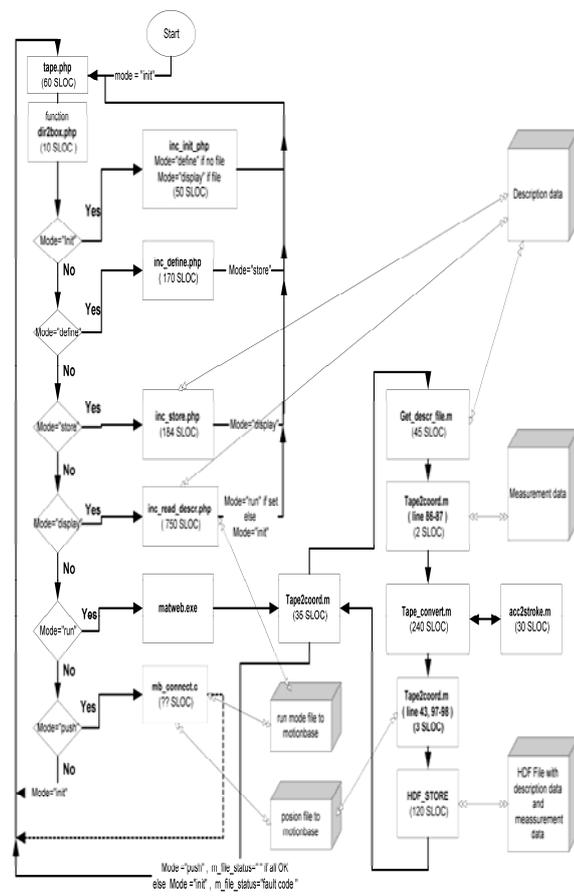


Fig 6 block diagram of module "Field Signal".

The definitions are stored by the web page and read from the MATLAB script as a structure data type. The only parameters transferred from the web-pages to the MATLAB script are name and location of the definition file and the paths where to store HDF and output files. As the recorded signals could come from accelerometers and be needed to be transformed to positions on the motion simulator, they need to be double integrated and HP-filtered (IIR) to take out offset drift. Next step are to combine signals to calculate the vehicles motions in 6 degrees of freedom at the position where the simulated object are located on the motion simulator. The motion signal was first sub sampled to the control loop frequency of the motion simulator and stored as a CSV file to be accessible from a web page, same as for the synthetic signal WBS-Service. We had to decide how to debug the MATLAB script, either to test each routine in the MATLAB command window or through MATLAB web server. As we had to include the field data definition structure into the debugging environment, we went for the web solution. Using this method and consider that MATLAB web server had very and sometimes fault error information, we used the old trick to comment non tested functions and lines combined with returning 5 debugging variables the result web page. We had to do another trick to return values from a MATLAB script to a PHP page as the MATLAB web server interprets html code of the returning web-page and substituting %variable\_name% by the content of that variable set in the MATLAB script and variables in PHP are defined as %variable\_name. Our trick was to include the variable\_name1=%variable\_name1% in a auto redirect meta command in normal html page to a PHP page.

**Remarks:** It was fairly easy to get the web server with the MATLAB web-server tool up and running by reading the installation notes. The code examples and templates supplied with the MATLAB web server tool box, helped us in coding the WBS-Servers.

We did not set up Multilanguage support in these WBS-Servers but it shouldn't be any problem modifying the code in this direction, possible by using the PHP class lib[ref metabase] to interact with any database. The same db could also store the field data description.

The real resource effort was less than 2 times the estimated effort using [ref. COCOMO], and that's well within the range of estimation in the early design phase. Most of this added time was spend on optimizing the MATLAB code to reduce the execution time, but there were also some functions added during the development that was mainly of logical nature needing a lot of design work.

And finally we had to develop the coding and debugging environment.

#### **Areas to develop:**

- In co-work with teachers and consultants make guides and routines on how to create WBS-Services and WBS-Portals according to for example experience Ref [gathany].
- Encourage classroom teachers to use the pedagogical possibilities with e-teaching and at the same time increase their personal qualification Ref [kirk]
- Add knowledge management tools and group workshop to form a WBS Portal, like searchable KB, forums and documentation library. There are a couple of them available from f.ex. Ref [Zend].

- Develop routines on how to convert a Femlab Ref. [FEMLAB] experiment to a WBS-Service.
- To improve the support for Multilanguage with for example db storage of language specific strings used by the script and editable trough FTP- or SOAP-connection.
- By creating the WBS Services consider ASTD's E-Learning Courseware Certification (ECC) Ref [ECC]. Note this certification focus mainly on technical issues.
- Adopt methods in Learning Objects (WBS Services) that will support Multiple Intelligence Ref [MI] in a user-oriented workflow learning process.
- To standardize the content and interface description of WBS-Services to support collection of a group of lessons and WBS-Services in compiling a class Ref [SCORM] and Ref [Dublin\_Core].

- Ref [B2B] B2B Simulation: Web Supported Simulation Service <http://www.b2bsim.de/>
- Ref [WBS] The WBSC and WebSim were presented at the "13. Conference of Simulation and Visualization in Magdeburg. 2002-03-01 Stefan Osterburg <http://www.b2bsim.de/~osterbur/simvis02/Simvis.ppt>
- Ref [HDF] Hierarchical Data Format (HDF) home page at the [National Center for Supercomputing Applications](http://hdf.ncsa.uiuc.edu/hdf4.html). <http://hdf.ncsa.uiuc.edu/hdf4.html>
- Ref [WBS Portal ] <http://wbsp1.dc.luth.se/wbsp.htm>
- Ref [SVIB] Hole body vibrations Development of simulators over 20 years where presented at the SVIB conference 2002; Sven-Erik Tiberg [http://beemer.dc.luth.se/SVIB\\_conf\\_02/Holebodyvibrations-filer/frame.htm](http://beemer.dc.luth.se/SVIB_conf_02/Holebodyvibrations-filer/frame.htm)
- Ref [PHP] PHP Hypertext pre-processor homepage <http://www.php.net>
- Ref [ZEND] Zend PHP Code gallery <http://www.zend.com/codex.php>
- Ref [MATLAB]: Mathworks Homepage <http://www.mathworks.com>
- Ref [EditPlus] Code Editor <http://www.editplus.com>
- Ref [SCROM] The Sharable Content Object Reference Model: <http://www.adlnet.org/index.cfm?fuseaction=scromabt>
- Ref [Dublin\_Core] Metadata Standard: <http://dublincore.org>
- Ref [FEMLAB] FEMLAB supplies highly sought-after new technology for the modelling and simulation of physics in all science and engineering fields. <http://www.euro.femlab.com/showroom/interactive/>
- Ref [MI] Using Multiple Intelligence Theory in the Virtual Classroom  
By Margie Meacham <http://www.learningcircuits.org/2003/jun2003/elearn.html>
- Ref [COCOMO] <http://sunset.usc.edu/research/COCOMOII/>
- Ref [ECC] Exemplars of the E-Learning Courseware Certification (ECC) Standards Standard One: Orientation; Darin E. Hartley and Ethan Sanders <http://www.learningcircuits.org/2003/may2003/ecc1.htm>
- Ref [Gathany] Scenario-Based E-Learning Model: A CDC Case Study; Nancy Gathany and Jeanette Stehr-Green <http://www.learningcircuits.org/2003/apr2003/gathany.htm>
- Ref [kirk] Exploring Careers in E-Learning; James J. Kirk <http://www.learningcircuits.org/2003/may2003/kirk.htm>
- Ref [pierce] Overcoming Obstacles to E-Learning: A Supplier's Perspective; Jack Pierce <http://www.learningcircuits.org/2003/jun2003/pierce.htm>
- Ref [purdue] Supplier Case Study: WebCT at Purdue University; Sarah Burke <http://www.learningcircuits.org/2003/may2003/purdue.htm>

- Ref [metabase] a PHP universal class library for databases  
[http://freshmeat.net/projects/metabase/?topic\\_id=914%2C68](http://freshmeat.net/projects/metabase/?topic_id=914%2C68)