

WEB-BASED TOOLS FOR SYSTEM DYNAMICS SIMULATION

Małgorzata ŁATUSZYŃSKA

University of Szczecin, Faculty of Economics and Management, Szczecin, Poland
e-mail: malgorzata.latuszynska@wneiz.pl

Abstract: A rapid development and omnipresence of the World-Wide-Web and its technologies have had remarkable impact on the field of computer simulation, understood as a numerical technique for conducting experiments with certain types of mathematical and logical models, describing the behavior of a system on a computer over extended periods of time. One of the computer simulation methods used to solve complex management problems is System Dynamics (SD). The aim of this paper is the review of SD Web-based tools in the context of some technology aspects of Web-based simulation and its advantages and disadvantages.

Keywords: System Dynamics, Web-based simulation

1 Introduction

The Internet and its multimedia front-end, the World-Wide-Web (WWW), has experienced remarkable growth since its introduction by Berners-Lee in 1989 (CERN, 2017). Many disciplines have re-evaluated their strategies, methods and techniques in the view of services offered by the Internet (Byrne et al., 2010). A rapid development and ubiquity of the WWW and its technologies have had also an impact on the field of computer simulation, understood as a numerical technique for conducting experiments with certain types of mathematical and logical models describing the behavior of a system on a computer over extended periods of time (Naylor et al., 1967). Kuljis and Paul (2001) go so far as to state that the pressure imposed by the increase of the Web uses has forced the simulation community to migrate to the Web in order to stay “alive”.

A Web-based simulation (WBS) can be defined as the use of resources and technologies offered by the Internet for ensuring an interaction with the client and server modelling and simulation tools (Byrne et al., 2010; Kuljis and Paul, 2001; Maciag et al., 2013). A common characteristic of all WBS applications is that they use a Web browser as a support for graphical interfaces connecting the user with simulation.

Although the notion of Web-based simulation is probably as old as the Web itself (Reichenthal, 2002), early WBS efforts began in 1995, first by providing Web-front ends to simulations running as Common Gateway Interface (CGI) scripts /programs. In addition, work began on Java-based simulation packages, systems and environments that would run anywhere on the Web. Results from these activities were first reported in the Winter Simulation Conference in 1996 (Buss and Stork, 1996; Fishwick, 1996; Kuljis, 1996; Nair and Miller, 1996). From this event, the interest and level of research effort in the area of Web-based simulation grew continuously, which is reflected by an increasing number of publications on this subject. Fig. 1 presents an upward trend of scientific works relating to Web-based simulation, published in ScienceDirect, in the decade of 1996–2015.

Most papers appeared in “Simulation Modelling Practice and Theory”. The reviewed papers discussed both the technical aspects of the Web-based simulation and its tools (Discrete Event Simulation, Multi-Agent Simulation or System Dynamics Simulation) as well as its application in various fields, such as medicine, education, computer games or decision support systems.

The aim of this paper is the review of System Dynamics (SD) Web-based tools in the context of some technology aspects of Web-based simulation and the advantages and disadvantages of WBS.

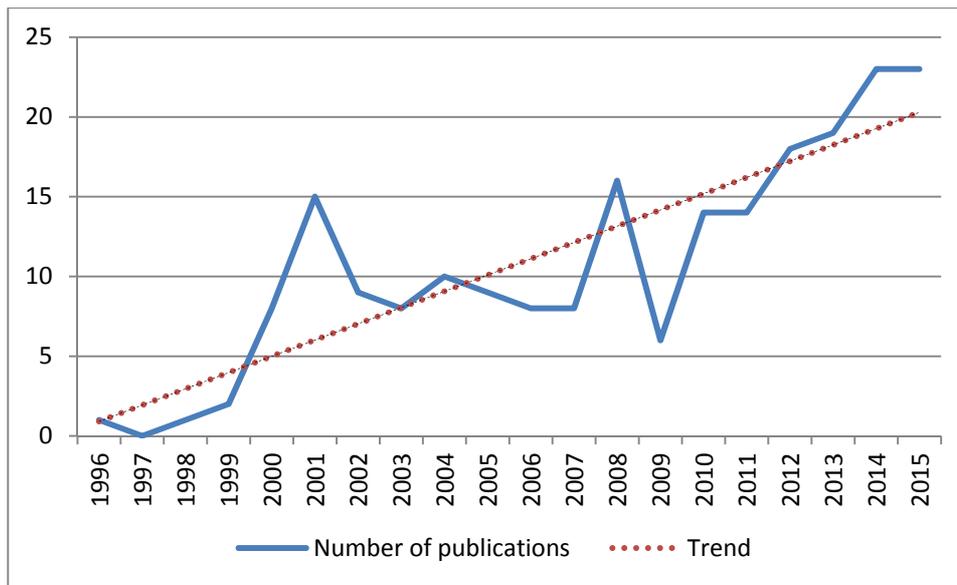


Figure 1. Number of publications relating to WBS in 1996–2015
(source: own study)

2 Categories and technologies of Web-based simulation

Relevant research suggests several categories of Web-based simulation, depending on the adopted criteria (for example, see Bencomo, 2004; Myers, 2004; Page, 1999; Whitman et al., 1998). This paper focuses on the types related to descriptions of how WBS applications can be developed architecturally, that is:

- local simulation and visualization,
- remote simulation and visualization,
- hybrid simulation and visualization.

Local simulation and visualization (S&V) means, that the simulation engine and visualization components are downloaded seamlessly by the client to the user's local computer, so that the graphical interface and the simulation engine coexist in the same environment (i.e., within the browser) shifting the responsibility for execution completely from the server to the client, making the server a central distribution point to the simulation but performing no real work (Byrne et al., 2010; Bencomo, 2004).

For the development of these simulations, Java is currently the only real possibility for providing a wholly independent hardware platform, although there is some study on the viability of using multimedia tools, such as Macromedia Flash, for the simulation of real processes (Bencomo, 2004).

Fig. 2 presents the basic local S&V configuration, after the initial loading phase. During this loading phase, the user opens a browser and navigates to a Web page typically containing an applet, and the browser automatically calls for the applet that is seamlessly downloaded to the client's computer. In this case, the applet contains both the simulation engine and the visualization/animation engine, which execute locally on the client-side. There may be a more advanced configuration in which, for example, a database might be present on the server-side (Byrne et al., 2010).

Remote simulation and visualization means that both the simulation engine and any animation generation engine are located and execute remotely, on the server-side (Bencomo, 2004; Holzinger et al., 2008). Access to these is through a browser on the client-side. This approach can be considered a job request on a batch processing system. Parameters are submitted to the simulation engine through the Web server, and results are returned to the user once the simulation has finished running (Myers, 2004). Communication between the graphical interface in the browser and the simulation engine is carried out using, for example, Common Gateway Interfaces (Morilla et al., 2001; V, 1996), sockets (Narayanan et al., 1999), Java remote method invocation (RMI), JavaBeans, Common Object Request Broker Architecture (CORBA), remote procedure call (RPC) or via front-end applications for simulator software (Byrne et al., 2010; Bencomo, 2004).

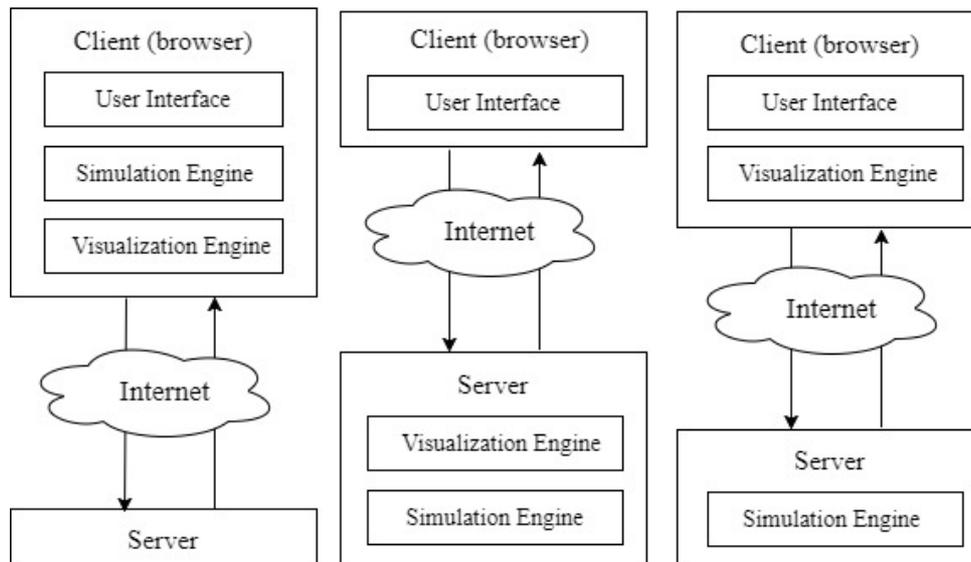


Figure 2. Basic configuration post initial loading phase
(source: own elaboration based on Byrne et al., 2010)

Fig. 3 shows the basic remote simulation and visualization configuration, after the initial loading phase. During this loading phase, the user opens a browser and navigates to a Web page and a basic interface is displayed in the browser. Both the simulation engine and any visualization/animation engine reside on the server-side, and are interfaced with through the browser. There may be a more advanced configuration in which, for example, a database might be present on the server-side (Byrne et al., 2010).

By combining the approaches of remote simulation and local visualization, hybrid simulation and visualization, can be created that yields the benefits of both (Miller et al., 2000; Myer, 2004; Whitman et al., 1998). In this approach, the simulation runs remotely on a simulation server, and when the user connects to the server through a WWW browser, a visualization/animation engine is downloaded to the client-side (Fig. 2) (Myers, 2004). A dedicated data connection may be established back to the server, in which the results of the simulation are transferred to the client-side allowing the visualization engine to display results to the user in a dynamic nature. This data can change continuously, delayed only by the executing simulation model and the latency present on the network connection. This approach is typically carried out by utilizing a Java server and Java applets (Myers, 2004), but may be carried out using other methods. Another implementation

of the hybrid simulation and visualization approach is where the visualization/animation engine is used on the client-side to “build” the simulation model through an animated interface, and display the results, without tying the animation to the simulation as the model is executing (Byrne et al., 2010).

Table 1 summarizes the basic characteristics of presented categories of Web-based simulation with a special focus on their advantages and disadvantages.

Finally, it is worth emphasizing that, apart from the architecture of WBS applications, the Web-based simulation is associated with such issues as (Byrne et al., 2010):

- model repository – which concerns the application of a server-based centralized repository that can be used to store simulation models. This allows for the ability to rapidly disseminate models, results and publications;
- Web-based simulation documentation – which relates to the efforts to provide on-line documentation to existing simulations including description(s) of modelling components, results of output analysis for a set of inputs, and animations displaying simulation modelling results. Such documentation can contain any or all the following elements: text, images, audio and video (Page, 1999).

Table 1. Comparison of WBS categories
(source: own study based on Byrne et al., 2010; Bencomo, 2004)

WBS category	Synonyms	Examples of applied technology	Advantages	Disadvantages
Local simulation and visualization	Local simulation and animation Local simulation visualization Local simulation Client-executed simulation	Java applets Macromedia Flash	The network latency between the user and the simulator is reduced to nothing.	The power and flexibility of the tool are minimized, as it relies on the power of the client to execute the simulations efficiently, and some users may lack the hardware needed for this.
Remote simulation and visualization	Remote simulation and animation based on loading applets Server-side-based Web simulation solution Remote simulation visualization Remote simulation Server-hosted simulation	Common Gateway Interfaces, Java remote method invocation (RMI), JavaBeans, Common Object Request Broker Architecture (CORBA), Remote Procedure Call (RPC) Front-end applications for simulation software	Larger simulations can run on powerful, high-end computers and the analysts can get access to the results from any low-end computer with a browser. In addition to providing a familiar interface, it works well for adapting existing simulation products to a Web-based environment. Easier maintenance for developers.	Not good for observing dynamic processes at work. It does not allow the user to interrupt a running simulation. The client can only view pre-specified outputs at a specified time. The developer requires a knowledge of middleware technologies. Long running simulations may take an extended time to generate output, especially under heavy load, resulting in the user having to wait long periods of time or having the client timeout by waiting too long for a server response. The server can easily be overloaded if all the responsibilities for simulation are placed on the server for several concurrent user simulations, or if there is a heavy need for, for example, interactivity, visualization data or numerical solvers.
Hybrid simulation and visualization	Animation and manipulation using a Java data server Remote simulation/local visualization Hybrid client/server simulation	Java server pages (JSP) Java applets ASP.NET	More powerful hardware. Maintenance is eased with a centralized application. The workload on the server is reduced. More user interaction by sending the animation/visualization portion to the client-side.	The communication between the client and server is delayed by network latency, which might be several seconds on a wide area network.

The issues discussed in this section are common for all simulation techniques. The differences in the processes of modelling and simulation conducted by means of varying simulation techniques are first of all a consequence of the very nature and the purpose of these techniques (Behdani, 2012; Borshchev, 2013). The sections below will present the substance of the System Dynamics simulation modelling and discuss the System Dynamics Web-based tools.

3 System Dynamics modelling and simulation

System Dynamics is the simulation modelling technique used mainly for the analysis of poorly structured problems, with numerous interrelations among elements. It originates from cybernetic approach to system analysis and allows for describing especially complex systems in the form of interactive and combinatory relations. Its theoretical foundations were created by Forrester and his colleagues from the Massachusetts Institute of Technology (Cambridge, MA, USA) in the '50's of the 20th century. Forrester's approach to industrial system modelling, the first application area for this method, is described in one of the earlier papers, entitled *Industrial Dynamics* (Forrester, 1971). Seven premises of the System Dynamics application are presented there:

- decision making process belongs to the class of control problems with information feedback,
- intuitive judgement of behavior of industrial systems is unreliable due to their complexity,

- experimenting on the model can facilitate understanding of anti-intuitive behavior of complex systems,
- cost of collection of data to carry out experiments on the model is relatively low,
- decision processes mapped by the model are close to real ones to such an extent that they allow for forming an appropriate policy,
- industrial system behavior is determined by its internal structure; therefore, most problems are internally generated,
- real system behavior is so detached from the ideal one, that each effort of improving the system behavior, is justified.

On the basis of the premises mentioned above, J.W. Forrester took a peculiar approach to modelling and analysis of system behavior, which is based on the system concept in the way that it is more closely related to practice than theory. J.W. Forrester considered System Dynamics as a facilitating tool for management of the highest level, in strategic planning, where traditional management theory is of limited effectiveness.

During above 60 years, the interpretation of System Dynamics underwent certain modifications, but it managed to preserve its theoretical assumptions, based on three scientific disciplines, that is, traditional management theory, cybernetics and computer simulation (Fig. 3).

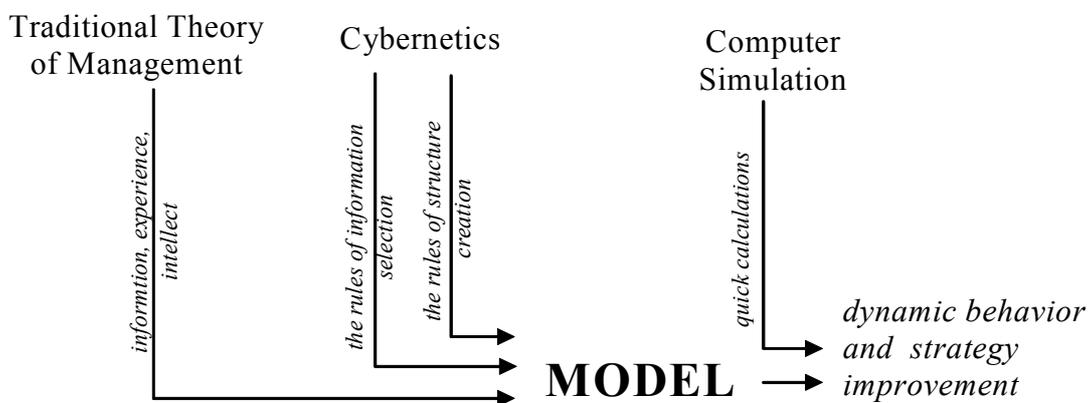


Figure 3. Methodological background of System Dynamics
(source: Łatuszyńska, 2005, p.32)

Traditional management is a process which consists of continuous decision-making. In a highly simplified perspective, this process begins from the observation of external world, identification of purposes, connections and information streams. As a result, imagined models are created, and the future behavior of systems in various conditions is predicted based on these models.

The most difficult part of this process is the selection of information and then finding the right course of action which should ensure that defined goals are met. Surprisingly, quite often, mistakes in management stem not so much from incorrect view of particular cause-and-effect chains, but from wrong understanding of system dynamics as a whole.

Cybernetics, which bases on feedback theory, puts a strong emphasis on dynamic interactions between system elements and provides tools allowing for better understanding of system behavior. It gives guidelines for distinguishing between important and unimportant information in a given context, and then for structuring and formalizing it in a mathematical model.

Finding solutions for such models with non-linear relationships (often made of numerous equations) and predicting the results of analyzed decisions by means of such models are generally only possible through use of appropriate numerical methods. In case of System Dynamics, the method of choice is computer simulation based on constant step method.

As a result, System Dynamics provides a rationally founded tool base for the construction of models, which should produce information about behavior of systems as a whole. Theoretical assumptions and detailed rules of System Dynamics modelling and simulation have been defined in multiple publications, including (Krupa, 2008; Meadows and Wright, 2008; Łatuszyńska, 2008).

4 Web-based tools for System Dynamics simulation

From the beginning of System Dynamics, many tools have been developed to support specific for this technique modelling of feedback loops, nonlinearity, and time delays that influence the behavior of the

entire system and to help to use this technique without necessary knowledge of mathematical background. The first tool was a special programming language called DYNAMO created in the 1960s. DYNAMO was a typical high-level language that was not quite friendly for users of low IT sophistication. Later, many other tools appeared on the market that now support the creation of System Dynamics models.

There is a constantly updated list of available tools on the System Dynamics Society web page (SDS, 2017). The list also contains tools that enable, to a greater or lesser degree, the creation and/or execution of System Dynamics models on the Web:

- AnyLogic by The AnyLogic Company (anylogic.com),
- Consideo iMODELER by Consideo GMBH (www.consideo-modeler.de),
- Forio Simulations by Forio Corporation (www.forio.com),
- Insight Maker, created by S. Fortmann-Roe and G. Bellinger, hosted by Give Team (www.insightmaker.com),
- SimiLive – service for Simile simulation software by Simulistics Ltd (similive.simulistics.com),
- Simupedia Studio - plugin for Simantics System Dynamics by VTT Technical Research Centre of Finland and Semantum Oy (sysdyn.simantics.org/; www.simupedia.com),
- Sysdea of Strategy Dynamics Ltd, (sysdea.com).

AnyLogic simulation software supports system dynamics, agent based and discrete event modeling. It allows making hybrid models. Since AnyLogic models are 100% Java applications, they can readily be published as web applets. With AnyLogic, the whole model will automatically run on the client machine within the applet. It is enough to send a ready-to-run simulation with GUI and experimentation capability to anyone with a browser. Many SD applets generated by AnyLogic are published on website www.runthamodel.com.

Table. 2 System Dynamics Web-based tools
(source: own study)

Tool name	Logo	Licensing	WBS category	Technology/ Language	Main features	Case studies available	On-line courses or documentation
AnyLogic		Proprietary, commercial, free Personal Learning Edition for education	Local	Java web applets	S&V of existing models with an option to enter input data	yes	yes
Consideo iMODEL-ER		Proprietary, commercial	Local	JavaScript Adobe Flash	Building, S&V of new models. S&V of existing models with an option to enter input data and make structural changes to the model	yes	yes
Forio Simulations		Proprietary, commercial	Hybrid	JavaScript, REST-based APIs Forio SimLang, Python, Julia, R, Vensim	S&V of existing models with an option to enter input data	yes	yes
Insight Maker		Free, Insight Maker Public License ^L	Local	JavaScript	Building, S&V of new models. S&V of existing models with a limited option to enter input data	yes	yes
SimiLive		Proprietary, commercial, Free evaluation license limited to 25 model elements	Hybrid	JavaScript C++, Prolog, Tcl	S&V of existing models with an option to enter input data	yes	no
Simupedia Studio		Eclipse Public license (EPL)	Hybrid	JavaScript Modelica	S&V of existing models with an option to enter input data	yes	yes
Sysdea		Proprietary, commercial, a 30-day free trial	Local	JavaScript	Building, S&V of new models. S&V of existing models with an option to enter input data and make structural changes to the model	yes	yes

Consideo iMODELER offers quantitative and qualitative modeling. This tool allows to build web-based models and is free for educational purposes. It runs on any modern PC, Mac, Smartphone or Tablet-Computer as a web-based application (iMODELER-Service) or alternatively as an installed App Launcher for Win, Mac or Linux. Sample models are available on the website: www.know-why.net.

Forio Simulations allows building web-based interfaces to already developed models, and provides a library of interactive environments for users. The software is web hosted, with a commercial use license, but provides a number of options for low cost and no cost publication and proliferation of models and model results. Sample models are available on the website: forio.com/simulate.

Insight Maker is a multi-method simulation and modeling platform that runs completely in a web browser. It supports causal loop diagrams, rich pictures, dialogue mapping, mind mapping, stock & flow simulation models. It enables to build, run and share System Dynamics models without downloading or installing a single program. Sample models are available on the website: insightmaker.com/tag/Sample-Model.

SimiLive is a web-based model execution service for Simile System dynamics software. It allows models built in the Simile modelling environment to be executed on a web browser. It provides a simple way to publish any model, including parameter data and visualization tool setup. The model diagram and visualization tools have been customized to drag and zoom via a touch screen interface, so the service is ideal for showing and executing models on mobile devices. Sample models are available on the website: similive.simulistics.com/demos.

Simupedia Studio is a plugin, which enables the generation of interactive Simupedia web pages. The power of Simupedia derives from the open source modelling and simulation platform called Simantics. The system has client-server architecture with a semantic ontology-based modelling database and Eclipse framework-based client software with plug-in interface. Sample models are available on the website: www.simupedia.com/Demo/#WorkRoom/Management.

Sysdea is an on-line, browser based, system dynamics tool. It is easy to use, has extensive documentation and easy model sharing options. Sample models are available on the website: strategydynamics.com/info/system-dynamics.aspx.

Table 2 includes additional information associated with the above-mentioned Web-based tools for System Dynamics modelling and simulation, specifying the used technology, the WBS form and the main features.

Only Consideo iMODELER, Insight Maker and Sysdea provide an opportunity to create both simulation models and their simulation and visualization via a browser. The remaining tools are first of all used for publishing in the Internet the models created by means of some simulation software. All of them allow changes in the model input data prior to the simulation. Additionally, some of the tools permit changes to the structure of the completed models (iMODELER, Sysdea).

Undoubtedly, the combination of System Dynamics simulation with WWW will help avoid many limitations typical of a traditional simulation constructed by means of a dedicated simulation software, especially when applied for educational purposes. The limitations mainly refer to: high outlays and considerable cost of simulation modelling tools (prices of core System Dynamics software, like iThink/Stella, Powersim Studio or Vensim range between 1 000 to over 7 000 euro, depending on the version), as well as the hermetic nature of the simulation expertise (which seems to be available only to experts and experienced users of simulation models (Maciag et al., 2013, p.314). On the other hand, WBS has its weaknesses – these are discussed in the next section of this paper.

5 Advantages and disadvantages of WBS

Web-based simulation has many benefits in comparison to the classical simulation systems, and many authors attempt to identify these, see (Miller et al., 2001; Bencomo, 2004; Kuljis and Paul, 2001; Maciag et al., 2013; Pidd and Carvalho, 2006; Whitman et al., 1998).

Byrne et al. (2010) lists for instance the following:

- Ease of use – the process of simulation model building is accepted in the simulation community to be a fundamentally difficult, time-consuming and error-prone one. Commonly known characteristics of the Web is its ease of navigation and use. The Internet provides a familiar interface for both interacting with and controlling a simulation. However, it seems to be very dangerous to make an extrapolation from the ease of use of common Web-based systems to WBS systems, as the typical work and necessary features are different.
- Collaboration – WBS environments can support collaborative model development, where people can communicate with each other from different places to build the same simulation model over the Web. Such collaboration can include monitoring and debugging with the help from experts. This approach may result in a reduction in simulation model development cost and time, over the use of System Dynamics simulation modelling packages such as Powersim, Stella/iThink or Vensim, which are only used at a local machine.
- Model reuse - modelers suggest that model reuse may in fact cost more than developing new models, as trust must be established through testing. The Web can support the reuse of existing simulation models very well by its distributed nature and the content management function like search machines and common data access protocols.
- Cross-platform capability - the Web allows to run an application on any Web browser on any operating system without compiling. This capability relieves the application developer from having to worry about a client's configuration.
- Controlled access - access can be controlled to a Web-based simulation application through the use of passwords, and limited time-span access can be allocated. Users who need access to specific or limited areas of an application can be given access merely by being added to a password list, instead of having their client machines updated.
- Wide availability - WBS application can be used from anywhere in the world with an Internet connection and outside of normal business hours without having to transport hardware or software.
- Versioning, customization and maintenance – in using a Web-based system, maintenance is minimized. All modifications can be made through the server, enabling frequent modifications, customizations and updates to be made and instantly distributed to the application, reducing error potential and eliminating virtually all on-site maintenance.
- Integration and interoperability – a Web-based tool can integrate and interoperate with both existing and future Web-based applications, as well as Web-enabled desktop applications.

Several disadvantages of WBS over classical simulation systems have also been identified. Byrne et al. (2010) lists, for instance, the following:

- loss in speed – the user can experience loss in speed when interacting with the tool due to downloading time and network traffic (Suh, 2005);
- graphical user interface limitation – the interface provided by the Web as opposed to desktop simulation tools is limited, although this is starting to change with the evolution of multimedia authoring tools for the Web,
- security vulnerability – Web-based applications are more vulnerable to malicious attacks than client-based applications,
- Web-based simulation application stability – the stability of a WBS application can be affected or disabled by, for example, the disappearance of any site that hosts parts of the modelling environment (Kuljis and Paul, 2001),
- licensing restrictions - if the simulation engine is developed using traditional proprietary simulation software, there may be license restriction issues, as traditional software licenses of simulation packages only allow a single place usage whereas a Web-based system must have the same license model as a network license; this can be overcome through the use of open source simulation software for the simulation engine.

The shortcomings listed above as well as the specific nature of computer simulation as a research method can constitute a barrier for further development of WBS.

As Kuljis and Paul (2001) state, it is the mismatch between the Web main characteristics and the ap-

proach taken by Web-based simulation so far, which has not changed from the classical approach to take advantages of the new media. This mismatch is illustrated well in the Table 3.

Table 3. Comparison between Web Features and Simulation
(source: Kuljis and Paul, 2001, p.47)

Web Features	Web - Based Simulation	Classical Simulation
Common Standards	Yes	No
Platform independence	Yes	No
Interoperability	Generally not supported	Not supported
Ease of navigation	Varies	Varies
Ease of use	Difficult to use	Difficult to use
No specialist knowledge required	Specialist knowledge required	Specialist knowledge required
Not affected by change	Affected by change	Affected by change
Unstructured	Structured	Structured

The Web is a truly democratic and non-exclusive medium as is evident by the exponential growth of Internet subscribers. The main reasons for this are that the Web is easy to use and to navigate through, is forgiving (one cannot make unrecoverable errors), and is easy to publish on. The access to the Web is widely available, does not depend on proprietary software, and so on. Simulation is and always was a highly specialist application area (Web-based or not) – not easy to use. Some simulation software does not have an obvious or consistent navigational strategy (Kuljis and Paul, 2001).

The stability of the Web is not affected by the addition of new sites, removal of existing sites, or any changes to sites. The stability of WBS, as it was mentioned before, can be easily affected or even disabled, for example, the disappearance of any site that hosts parts of modelling environments (Kuljis and Paul, 2001).

The Web is a vast collection of often unstructured information that is always available and accessible (as long as the server holding it is 'alive'), regardless

of the format it is in (text, pictures and hypermedia). Simulation uses data that is not generally accepted in a standard format and usually has to be provided in a specific format (Kuljis, 1996). Modelling components and rules of assembling these components also have to follow standards specific for simulation software. Outputs from the simulation again follow different rules and vary in: what is reported, how it is reported, at which stage of the simulation run it is reported, formats of output files, and so on. (Kuljis and Paul, 2001).

6 Conclusion

The World-Wide-Web enables a distributed environment for utilizing simulation tools. By using modern Web technologies, the transition to Web-based simulation is simplified. As the above short review has shown, much has been done in the area of WBS – also in the context of System Dynamics simulation technique.

But despite a whole range of opportunities and the prevalence of the WWW, there seems to have been rather little impact on business users. It is hard to find WBS applications reported in the business context as the educational context prevails.

Despite the above-mentioned faults and barriers to the widespread use of WBS, it seems that this direction of the development of computer simulation tools still offers considerable potential. Presently, it aims at broadened application of cloud computing that provides various IT resources to users transparently in an easy-to-use, on-demand, cheap and pay-as-you-go manner. It frees users of burdens associated with managing computing resources and facilities, and reduces or even eliminates the capital outlays in hardware. These advantages are helpful to make simulation techniques become more accessible to users (Kothari et al., 2015; Liu et al., 2012). Cloud-based computer simulation is becoming more and more attractive to modelling and simulation practitioners.

7 References

- [1] Behdani, B., 2012. Evaluation of paradigms for modeling supply chains as complex socio-technical systems. In: *Proceedings of the 2012 Simulation Conference WSC*, Huntington, pp.1-15.
- [2] Bencomo, S.D., 2004. Control learning: present and future. *Annual Reviews in Control*, 28(1), pp.115-136.
- [3] Borshchev, A., 2013. *The big book of simulation modelling*. North America: AnyLogic.
- [4] Buss, A., Stork, K., 1996. *Discrete event simulation and Worlds-Wide-Web using Java*. In: J.M. Charnes, D. J. Morris, D.T Brunner, J.J. Swain eds., 1996. Proceedings of the 28th Conference on Winter Simulation, Coronado, pp.780-785.
- [5] Byrne, J., Heavey, C., Byrne, P.J., 2010. A review of Web-based simulation and supporting tools. *Simulation Modelling Practice and Theory*, 18, pp.253-276.
- [6] CERN, 2017. *The birth of the web*. Centre for European Nuclear Research, <http://home.cern/topics/birth-web> [Accessed 25 June 2017].
- [7] Fishwick, P.A., 1996. *Web-based simulation: some personal observations*. In: J. M. Charnes, D.J. Morris, D.T Brunner, J.J. Swain eds., 1996. Proceedings of the 28th Conference on Winter Simulation, Coronado, pp.772-779.
- [8] Forrester, J.W., 1961. *Industrial Dynamics*. New York: Wiley.
- [9] Holzinger, A., Emberger, W., Wassertheurer, S., Neal, L., 2008. Design, Development and Evaluation of Online Interactive Simulation Software for Learning Human Genetics. *Elektrotechnik & Informationstechnik*, 125(5), pp.190-196.
- [10] Kothari, S., Peck, T., Zeng, J., Oblea, F., Votaw, A., E, Dispoto, G., 2015. Simulation as a Cloud Service for Short-Run High Throughput Industrial Print Production Using a Service Broker Architecture. *Simulation Modelling Practice and Theory*, 58, pp.115-139.
- [11] Krupa, K., 2008. *Modelowanie symulacja i prognozowanie. Systemy ciagle*. Warszawa: Wyd. Naukowo-Techniczne.
- [12] Kuljis, J., 1996. *HCI and Simulation Packages*. In: J. M. Charnes, D.J. Morris, D.T Brunner, J.J. Swain eds., 1996. Proceedings of the 28th Conference on Winter Simulation, Coronado, pp.687-694.
- [13] Kuljis, J., Paul, R.J., 2001. An Appraisal of web-Based Simulation: Whither We Wander? *Simulation Practice and Theory*, 9(1-2), pp.37-54.
- [14] Łatuszyńska, M., 2005. *Modelling of Consequences of Transport Corridors Development*. LIBERTAS Paper 62, November.
- [15] Łatuszyńska, M., 2008. *Symulacja komputerowa dynamiki systemów*. Gorzów Wielkopolski: Wydawnictwo PWSZ.
- [16] Liu, X., He, Q., Qiu, X., Chen, B., Huang, K., 2012. Cloud-based Computer Simulation: Towards Planting Existing Simulation Software into the Cloud. *Simulation Modelling Practice and Theory*, 26, pp.135-150.
- [17] Maciąg, A., Pietroń, R., Kukła, S., 2013. *Prognozowanie i symulacja w przedsiębiorstwie*. Warszawa: PWE.
- [18] Meadows, D.H., Wright, D., 2008. *Thinking in Systems: A Primer*. Chelsea: Chelsea Green Publishing Company.
- [19] Miller, J.A., Fishwick, P.A., Taylor, S.J.E., Benjamin, P., Szymanski, B., 2001. Research and Commercial Opportunities in Web-Based Simula-

- tion. *Simulation Practice and Theory*, 9(1-2), pp.55-72.
- [20] Miller, J.A., Seila, A.F., Xiang, X., 2000. The JSIM Web-Based Simulation Environment. *Future Generation Computer Systems*, 17(2), pp.119-133.
- [21] Morilla, F., Fernández, A., Dormido-Canto, S., 2001. Control Systems Analysis and Design Server. In: S. Dormido, F. Morilla, eds., 2001. *Internet Based Control Education 2001: A Proceedings Volume from the IFAC Workshop*. Pergamon: Madrid, pp.151-156.
- [22] Myers, D.S., 2004. *An Extensible Component-Based Architecture for Web-Based Simulation Using Standards-Based Web Browsers*. Blacksburg: Department of Computer Science, Virginia Polytechnic Institute and State University.
- [23] Nair, R., Miller, J.A., 1996. *A Java-Based Query Driven Simulation Environment*. In: J. M. Charnes, D.J. Morris, D.T Brunner, J.J. Swain eds., 1996. Proceedings of the 28th Conference on Winter Simulation, Coronado, pp.786-793.
- [24] Narayanan, S., Rao, N., Geist, J., Kiran, P., Ruff, H.A., Draper, M., 1999. UMAST: A Web-Based Architecture for Modeling Future Uninhabited Aerial Vehicles. *Simulation*, 73(1), pp.29-39.
- [25] Naylor, T.H, Burdick, D.S, Sasser, W., 1967. Computer Simulation Experiments with Economic Systems: The Problem of Experimental Design. *Journal of the American Statistical Association*, 62(320), pp.1315-1337.
- [26] Page, E.H., 1999. *Beyond Speedup: PADS, the HLA and Web-Based Simulation*. In: R. Fujimoto, S.J. Turner, eds., 1999. Proceedings of the Thirteenth Workshop on Parallel and Distributed Simulation, Atlanta, pp.2-9.
- [27] Pidd, M., Carvalho, A., 2006. Simulation Software: Not the Same Yesterday, Today or Forever. *Journal of Simulation*. 1(1), pp.7-20.
- [28] Reichenthal, S.W., 2002. *Re-introducing Web-Based Simulation*. In: E. Yücesan, C.-H. Chen, J.L. Snowdon, J.M. Charnes, eds., 2002. Proceedings of the 34th Conference on Winter Simulation: Exploring New Frontiers, San Diego, pp.847-852.
- [29] SDS, 2017. *Tools for System Dynamics*. *System Dynamics Society*, <http://tools.systemdynamics.org/> [Accessed 25 June 2017].
- [30] Suh, W., 2005. Web Application Development Methodologies. In: J. Travers, ed., 2005. *Web Engineering: Principles and Techniques*. London: Idea Group Publishing, pp.76-93.
- [31] Whitman, L., Huff, B., Palaniswamy, S., 1998. *Commercial Simulation over the Web*. In: D.J. Medeiros, E.F. Watson, J.S. Carson, M.S. Manivannan, eds., 1998. Proceedings of the 30th Conference on Winter Simulation. Washington, pp.335-340.