USING THE VAADIN WEB FRAMEWORK FOR DEVELOPING RICH ACCELERATOR CONTROLS USER INTERFACES*

Wenge Fu[#], Kevin Brown, Ted D'Ottavio, Seth Nemesure, Enrique Schuhmacher Brookhaven National Laboratory, Upton, NY 11793, USA

Abstract

Applications used for Collider-Accelerator Controls at Brookhaven National Laboratory typically run as console level programs on a Linux operating system. One essential requirement for accelerator controls applications is the bidirectional synchronized IO data communication. Several web frameworks have made it possible to develop web based Accelerator Controls applications that provide all the features of console based user interface applications. Web based applications give users flexibility by providing an architecture independent domain for running applications. Security is established by restricting access to users within the local network. Additionally, the web framework provides the opportunity to develop mobile device applications that makes it convenient for users to access information anywhere and anytime. The Vaadin Java Web Framework is a tool kit being used to develop client side web interfaces. Vaadin provides Java developers a short learning curve overhead. Most Java Technologies, including JavaEE and third party packages work well within the Vaadin framework. This paper explores the feasibility of using the Vaadin web for developing UI applications framework for Collider-Accelerator controls at Brookhaven National Laboratory.

INTRODUCTION

"Vaadin Framework is a Java web application development framework that is designed to make creation and maintenance of high quality web-based user interfaces easy"[1]. First released in 2009, the Vaadin web application framework has been a fast growing API in terms of popularity among web developers for its rich functionality. The Vaadin framework has an advantage over other web development technologies because it uses the Java programming language which is more familiar to the application development community.

The Vaadin application framework provides two programming models: server side (Java) and client side. The client side framework is backed by the Google Web Toolkit (GWT). Program code is written in Java and resides on the server side. Server side program code helps make web applications more secure. Vaadin has a rich set of UI components. The server side and client side communicates via HTTP (or TCP when websockets are used) protocol and transfers data in JSON. The Vaadin

* Work supported by Brookhaven Science Associates,

framework supports all major web browsers without additional plugins[2] and works well with major IDEs. This makes the web application coding and debugging easier. In Vaadin, the look and feel of the web application is controlled by CSS themes. This makes web application GUI richer, and more configurable; Vaadin is best used for designing single page web applications which typically work like console level UI applications.

VAADIN FOR ACCELERATOR CONTROL APPLICATIONS

Controls applications used in accelerator controls systems have many common characteristics, such as:

- They mostly require fast live bidirectional communications with many different systems such as hardware controllers, database servers, file servers, and other legend systems on different platforms (Unix, Linux, Windows etc.);
- Requires fast UI and interactive responsiveness.
- Rich UI for control data visualization for single or multiple GUIs.

These features can be relatively easy to implement with traditional languages such as C++ and Java. As Vaadin uses the Java programming language, server side java JDK (or JavaEE) APIs, third party APIs and jar packages can be used directly. This makes Vaadin a favorable choice when choosing a web application framework.

In a Vaadin web application, the server side programs (written by developers) and client side code (generated by Vaadin from server side code) have a common shared state, which helps enhance UI responsiveness and overall performance. For web based accelerator controls applications, the UI design and GUI layout are relatively easy to implement. It is critical for the client-server bidirectional communication layer to be effectively managed. Vaadin data push features make this kind of bi-directional communication easy to setup.

Vaadin push can be configured with Java annotations: @Push(PushMode, Transport) in program code or with a configuration file. There are three push modes:

- AUTOMATIC (default)
- MANUAL
- DISABLED

and 3 transport methods for communication in the request/response cycles:

- LONG_POLLING
- STREAMING
- WEBSOCKET

ISBN 978-3-95450-148-9

LLC under Contract No. DE-SC0012704 with the U.S.

Department of Energy.

[#] fu@bnl.gov

These push modes and transport methods can be either set in the program or set in a configuration file (e.g. web.xml).

is Since STREAMING verv similar to LONG POLLING and has been deprecated since V7.5.0, this paper just focuses on the long_polling (the "fake" push) and websocket (the real push) methods.

Long polling is a process whereby clients send requests to the server, and the server receive the requests. but holds it for a set period of time. The response happens when new data is available within the time period. Meanwhile, the client keeps the connection open and ready to receive data from the server. In this method, although all requests are initiated from client side, it effectively achieves a real time server push-like bi-directional communication.

Websocket, on the other hand, is a full duplex TCP connection that is independent of the HTTP request/response cycle. Once the connection is established, it is a true real time bi-directional communication.

For accelerator controls applications, both methods work well. But they do have some differences because of the nature with which the communication is established. The long polling method is easy to setup and is supported by most web browsers. However, there may be a short delay in data transport, while requiring more server resources such as memory. Websockets use dedicated connections between server and clients, use fewer resources and is capable of handling large amounts of client/server connections. Figure 1 diagrams the relationship between web application (client), Vaadin application server and the back end accelerator control system.

VAADIN CONTROL APPLICATION **EXAMPLES**

Vaadin is primarily designed for single-page web applications which work like desktop applications. The common life cycle for developing Vaadin web applications include, designing and writing web application UI Java code, tuning the look and feel of the UI in CSS, deploying the application to a web container (such as Glassfish server) and testing the application in a web browser. Since the UI design is very similar to strategies used with other languages, the focus of our tests addressed synchronized IO communication with simple GUIs and default CSS settings. We tested Vaadin with 3 different types of controls applications:

- 1. A single page application with no push (data polling, HTTP).
- 2. A single page application with manual push in long polling. (HTTP)
- 3. A single application with automatic push in websockets. (TCP)

The Glassfish server v4.1 (and v4.0) are used as the application servers.

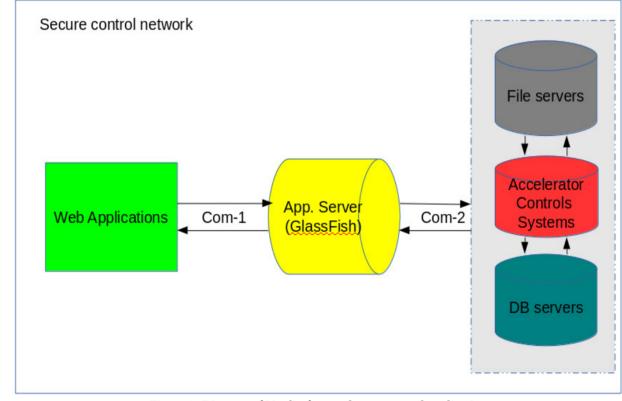


Figure 1: Diagram of Vaadin for accelerator control application

In the first case, we converted a C++ based controls application into a Vaadin web application. The application's name is SystemViewer. The basic function of this simple program is to display the current statuses of all monitored control systems based on live data in a back end database, and to highlight any system which may have problems or need System Administrators' attention. This program also displays the detailed system data for any monitored systems, and is capable of launching <100Hz. When the frequency > 100Hz, the GUI becomes sluggish. On a graphical display, a frequency >10 Hz is usually not necessary. In practice, this kind of high frequency push may not be reasonable for GUI applications. Figure 3 shows this application in C++ vs Vaadin version, Tests concluded that, the performance difference between data push with long_polling and websockets are negligible. Both long_polling and websockets can be used in controls application

			S	stemViewer							
File Options											Help
isplay Mode: All with Bad S	Status 💷	Update Every: 10	seconds	Update: 🔷 O	on ⇔ off	Refresh Da	ita	Last Update	d @ Mon Au	ug 24 11:29:	12 20
System Statuses Reported h	y System Monitors										
Object Name		Status	Los	. Checked	Last	Alarm	Last (Aut	o)Started		MonitorNa	me
cfe-9c-ps1		se failure -> ac				3 06:15:26			Enabled		
SMacnlind3 System Monit Toshiba Siemens UPS	not ru	nning -> acnlind				2 10:31:00 8 13:02:17	Fri Jan 2	06:41:43		Master	
1006b-controls-ups	UPS to			24 11:26:59						Inspector	
Alcove 9C UPS-3	UPS tr	ouble		24 11:26:43						Inspector	
Narning Swap Space - Lo		se failure → cs				4 10:00:38				Auxiliary	
fe-930-time2 PS937BMan_56Mhz		se failure -> ac nning -> csinjec				4 16:17:36	Two Arms 11	15.00.00		No Longer No Longer	
-355 rbhan_30hhz	proc ru	uning -> carulec	colved Hug	12 15:04:10	Med Hog I	2 00:20:22	The Hug II	15:25:50	chabled	no conger	MC
ସ											×
SystemViewer updates data ∉	every 10 seconds.										
			System V	iewer - Mozilla Fir	refox						
	arks Tools Help	Custom Minute	a) o								
	Book of Vaadin - vaa ×	System viewer	* 1 m				Concerning and Concer				
Sadin API Index ×		System viewer	* (<u>4</u> ,			• e q.	Search	• 1	n 🛧 🖨	© ∢ e	> =
				ystem S	Statu		Search	• 1	ê 🖈 🖨	0 4 6	• =
🐑 🕲 acriline3.pbn.bnl.gov 8060	/SystemVlewer/		trol S	ystem S				් 1 ය@: 08/24/15 1		♥ 4 € rows 12 1	
	/SystemViewer/ atus	Con	trol S			ses					
	/SystemViewer/ atus	Con	trol S			ses	Last Update			rows 12	t Help
G acriline3.pbn.bnt.gov 8080 Display Mode: All with Bad St. System Statuses Reported by Syst	/SystemVlewer/ atus v emMonitors	Con Update Every: 10	seconds	Update: Cn	o or 💽	SES 3 Refresh Data	Last Update	යල: 08/24/15 1	11:31:21	rows 12	t Help
O acriline3.pbn.bnt.gov 8080 Display Mode: All with Bad St. System Statuses Reported by Syst Object Name	/SystemVlevier/ atus v emManitors stotus	Update Every: 10	seconds	Update: Cri Last Norm	o orr 💽	SES 3 Refresh Data	Last Update	යල: 08/24/15 1	11:31:21 MonitorNa	rows 12 1 me Code	
actiline3.pbn.bnl.gov 9000 actiline3.pbn.bnl.gov 9000 lisplay Mode: All with Bad St. System Statuses Reported by Syst Doject Name 10060-controls-ups	/System Viewer/ atus v emManitors Stotus UPS trouble UPS trouble UPS trouble mot running >	Update Every: 10	20:03 2015 29:/7 2015	Las: Nam Tue Aug 18 13:0	Off Corr C	SES 3 Refresh Data	Last Update	යල: 08/24/15 1	Monito Nar Inspector	rows 12 1 me Code	t Help
	/System/Viewer/ atus v emManitors Status UPS trouble UPS trouble DPS trouble not rouning > calinget01 pbn.bnl.gov	Update Every: 10 Lass Checked Mon Aug 24 11: Mon Aug 24 11:	seconds :20:08 2015 :29:47 2015 :04:10 2015	Update: On Last Nam Tue Aug 18 18:0 Tue Aug 18 13:0	• om •	Ses 2 Refresh Data	Last Update	d@: 08/24/15 1 Nami/	Monito Nar Inspector Inspector	rows 12 1 me Code	P Help
Display Mode: All with Bad St. System Statuses Reported by Syst Display Mode: Display Mode System Statuses Reported by Syst Display Mode. System Statuses Reported by System Display Mode. System Statuses Reported by System System Stat	/SystemViewer/ emManitars Status UPS trouble UPS trouble UPS trouble UPS trouble calingcc0/pbn.bnl.gov renimd3.ppn.BNL.Gov pot runping.>	Update Every: 10 Last Checked Mon Aug 241 11 Mon Aug 241 11 Weed Aug 12 13	20:03 2015 29:47 2015 30:03 2015 30:03 2015	Last Nam Tue Aug 18 13:0 Tue Aug 18 13:0 Wed Aug 12 00	Off: 18 2015 04:53 2015 :28:22 2015 :31:00 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update	d@: 08/24/15 1 Nami/	MonitorNa Inspector No Longe	rows 12 1 me Code	P Help
	system Viewer/ emManitars status UPS trouble UPS trouble out running School gov and running School gov and running School gov	Update Every: 10 Update Every: 10 Ust Checked Mon Aug 241 11: Wed Aug 12 13 Mon Aug 24 11:	20:03 2015 29:47 2015 29:47 2015 30:03 2015 30:03 2015 30:16 2015	Update: On Last Nam Tue Aug 18 13:0 Tue Aug 18 13:0 Wed Aug 12 10 Wed Aug 12 10	orr 5 05:18 2015 04:53 2015 :28:22 2015 :31:00 2015 :06 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:22:38 2015 5:41:43 2015	dæ: 08/24/15 1 Nam/ Enabled	Monito Mar Inspector Inspector No Longer Mester	rows 12 1 me Code	P Help Ph Ph Sy
	system Viewer/ atus emManitars status UPS trouble UPS trouble out running s entimda pish.abnt.gov nat running s entimda pish.abnt.gov nat working response failure >>	Update Every: 10 Las: Checked Mon Aug 24 11: Week Aug 12 13 Mon Aug 24 11: Mon Aug 24 11:	20:08 2015 29:07 2015 29:47 2015 30:03 2015 30:03 2015 30:03 2015 30:059 2015	Lupdate: Cri Las: Nam Tue Aug 18 13:0 Tue Aug 18 13:0 Wed Aug 12 00 Wed Aug 12 10 Sut Aug 8 15:32	05:18 2015 04:53 2015 :28:22 2015 :31:00 2015 :06 2015 :06 2015 :02:17 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:22:38 2015 5:41:43 2015	dæ: 08/24/15 1 Nam/ Enabled Enabled	Monitoritia Inspector Inspector No Longe Mester No Longe	rows 12 1 me Code	P Help Ph Ph Sy
	system Viewer/ atus emManitors Status UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble other unning shill go the status of the	Logate Every: 10 Las: Checked Mon: Aug 24111: West Aug 24111: West Aug 24111: Muni: Aug 2411:	20:03 2015 29:47 2015 29:47 2015 30:03 2015 30:03 2015 30:03 2015 30:59 2015 29:17 2015	Lupdate: On Lass Nam Tue Aug 18 13:0 Wed Aug 12 10 Sot Aug 8 15:32 Tue Aug 18 13:0	05:18 2015 04:53 2015 :28:22 2015 :31:00 2015 :06 2015 02:17 2015 :00:38 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:23:38 2015 5:41:43 2015	dæ: 08/24/15 1 Nam/ Enabled Enabled	H1:31:21 MontorNat Inspector Inspector No Longe Mester No Longe Inspector	rows 12 1 me Code r Wats r Wats	P Help Ph Ph Sy
	system Viewer/ atus trimManitors Status UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble Calinget01, pbn, bnl.gov aching3, pbn, BNLgOV aching3, pbn, BNLgOV aching5, pbn, BNLgOV response failure -> response failure ->	Logate Every: 10 Las: Checked Mon: Aug 24111: West Aug 24111: West Aug 24111: Muni: Aug 2411:	20:08 2015 29:47 2015 29:47 2015 30:03 2015 30:03 2015 30:118 2015 30:118 2015 30:118 2015 29:17 2015	Last Nam Last Nam Tue Aug 18 13:0 Wed Aug 12 10 Sat Aug 81 13:0 Ute Aug 18 13:0 Ute Aug 18 13:0 Tue Aug 18 13:0 Mon Aug 24 10	05:18 2015 04:53 2015 :28:22 2015 :31:00 2015 :06 2015 n2:17 2015 :00038 2015 s1:02 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:23:38 2015 5:41:43 2015	dg: 08/24/15 t Nam/ Enabled Enabled Enabled	H1:31:21 Monto-Mar Inspector Inspector No Longe Inspector Ausiliary	rown: 12 1 me Code r Wats r Wats	P Help Ph Ph Sy
	system Viewer/ atus emManitars Status UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble UPS trouble acting s^ scinget01; jbn.bnl.gov acting s^ csinget01; jbn.bnl.gov acting sponse failure -> csioget03 response failure -> csioget04 response failure -> csinget34, pbn.bnl.gov	Last Checked Mon Aug 24 11: Mon Aug 24 11: Wed Aug 12 13: Mon Aug 24 11: Wed Aug 12 13: Mon Aug 24 11:	seconds 20:08 2015 20:47 2015 30:03 2015 30:03 2015 30:03 2015 30:59 2015 29:17 2015 29:17 2015 29:17 2015 2:21 2014	Last Nam Tue Aug 18 13:0 Wed Aug 12 10 Wed Aug 12 10 Sut Aug 8 15:22 Tue Aug 18 13:0 Mon Aug 24 10 Wed Aug 5 08:3	05:18 2015 04:53 2015 :28:22 2015 :28:22 2015 :31:00 2015 :08 2015 02:17 2015 :00:38 2015 :00:38 2015 :00:38 2015 :21:22 2015	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:23:38 2015 5:41:43 2015	dg: 08/24/15 t Nam/ Enabled Enabled Enabled	Monto-Mar Inspector No Longet No Longet No Longet No Longet No Longet No Longet	rown 12 1 me Code r Wat: r Wat: r Wat: r Wat:	P Help
	system Viewer/ atus emManitars status UPS trouble UPS trouble tot running entinda.phon.BNL.GOV nat running entinda.phon.BNL.GOV nat running entinda.phon.BNL.GOV nat running estinget3.phon.BNL.GOV response failure estopget05	Update Every: 10 Update Every: 10 Update Every: 10 Use: Checked Mon Aug 2411: Weed Aug 1213 Mon Aug 2411: Mon Aug	seconds seconds :20:08 2015 :29:47 2015 :30:03 2015 :30:03 2015 :30:03 2015 :30:39 2015 :29:17 2015 :29:17 2015 :29:17 2015 :29:17 2015 :29:17 2015 :29:17 2015	Update: On Las: Nam Tue Aug 15 13:0 Tue Aug 12 10:0 Wed Aug 12 10 Sa: Aug 8 15:12 Tue Aug 16 13:0 Wed Aug 12 10 Sa: Aug 8 15:12 Tue Aug 18 13:0 Wed Aug 12 10 Sa: Aug 8 15:32 Tue Aug 18 13:0 Mon Aug 24 10 Wed Aug 5 15:32 Wed Aug 5 10:32 Wed Aug 5 12:02	or 53 2015 04:53 2015 128:22 2015 128:22 2015 128:22 2015 120:2015 120:2015 120:2015 11:02 2015 21:21 2014 12:29 2014	Last (McCO) Tue Aug 11 Fri Jen 2 00	Last Update cared 1 15:23:38 2015 5:41:43 2015	de: 08/24/15 1 Nam/ Enabled Enabled Enabled Enabled Enabled	Montoniar Inspector Inspector No Longe Mester No Longe Inspector Auxiliary No Longe No Longe	rown 12 1 me Code r Wat: r Wat: r Wat: r Wat: r Wat:	P Hell

Figure 2: SystemViewer application: C++ (top) vs Vaadin (bottom)

diagnostic tools (other GUI programs) with related context directly from the web application. All features of the C++ version were implemented in the Vaadin version. This makes the program available anywhere within the security network. In this case, all data are periodically polled from the client side and the program works just as fast and as reliable as the C++ version. In fact, the Vaadin version includes more features such as allowing hiding and showing columns in the GUI. Figure 2 shows the GUI of the C++ vs the Vaadin web GUI.

In the second and third cases, we converted a GUI instance of a C++ application called PET (Parameter Editing Tool) into a Vaadin web application and use the data push approach with long_polling and websockets, respectively. In these two cases, the programs connect to control devices and display (or change) the live setting or measurement values of these devices. Two of the devices have data updating at various frequencies ranging from 1Hz to 1000Hz. In both cases, we found that the Vaadin web application works well when the data frequency development. It is recommended that websocket based push is used for long running web applications(>24

hours). Long_polling is supported by most of existing systems as it is an HTTP based "fake" push technology. Websocket is a relatively new technology, and requires new versions of application server software. Websockets are the recommended push technology for web based control applications.

In our Control System, we have successfully developed a DashBoard web application system using the Vaadin technology. It provides a flexible system for quickly setting up web based controls applications with a rich GUI to monitor accelerator operations.

Testing has uncovered some common problems with data push in Vaadin:

- After an application is running for some period of time, a "UIDetachedException" error occurs causing the web application to stop updating IO data. When this happens, re-loading page doesn't always help. The application server requires a restart. This problem may be caused by user session time outs.
- The Web UI seems unable to handle high frequency IO data with data update rates >

	FECs/Managers/spec	Fran] —
age PPM Device	Data Tools Buffer		He
ADO	D 1 D 1 D 1	T. 1	
ngSpec,blue	Ring State Ion Species	Injection Tuning PP	5
ngSpec.blue	Beam Energy	0.94	
ngSpec.blue	Gamma	1.00	
ngSpec.blue	Stone Type	1.00	
ngSpec.blue	Momentum Spread		
ngSpec.blue	Synchrotron Tune		
ngSpec.blue	Vertical Chromaticit		
ngSpec.blue	Horizontal Chromatici		
ngSpec.blue	Fill Number	19275	
ngSpec,blue	Time Of Fill Start		
ngSpec,blue	Store Length (mins) Time Of Flattop Star		
ngSpec.blue	Rigidity from ADO name		
-8-1			
(5,1) "text"		Nudge: 0 🖤 📠	1
(-,-,-,			-
-	5: Get and Async requests comple		
Edit View History	apecMan (Web Version - Mos Bookmarks Ibols Help * \-0-	zilla Firefox) —
Edit View History	apecMan (Web Version - Mos Bookmarks Ibols Help * \-0-		
RHIC Ring	Bookmank (Web Version - Moo Bookmarks Bools Help * \ 4 * (4 scMan2/ g Status	zilla Firefox	
RHIC Status By spech	apecMan (Web Version - Mox Bookmarks Book Help * + + ecMan2/ g Status	▼ C Q. Search ≫	
RHIC Status By specM	specMan (Web Version - Mos Bookmarks Bools Help * \ + ecMan?/ g Status	▼ C C Search >>	
RHIC Rin; ADO ringspec.blue	Bookmarks Book Version - Mox Bookmarks Book Help * \ 4 scMan2/ g Status lan Description Ring State	▼ C C Search ≫	
RHIC Status By speck	apecMan (Web Version - Mox Bookmarks Tools Help * + ecMan2/ g Status lan Description Ring State ion Species	Values Injection Tuning PP	
RHIC Status By speck ADO ringSpec.blue ringSpec.blue ringSpec.blue	Bookmarks Book Version - Max Bookmarks Book Help * 4 ecMan2/ g Status tan Description Ring state Ion Species Beam Energy	Values Injection Tuning PP 0.938272013	
Edit Miew History accilines.10000/pr RHIC Ring RHIC Status By speck ADO ringSpec.blue ringSpec.blue ringSpec.blue	Bookmarks Book Version - Mox Bookmarks Book Help * 4 scMan2/ g Status tan Description Ring State Ion Species Boom Encogy Gamma	Values Injection Tuning PP	
RHIC Status By speck ADO IngSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	apecMan (Web Veraion - Mox Bookmarks Tools Help * • ecMan2/ g Status lan Description Ring State Ion Species Beam Energy Gamma Stone Type	Values Injection Tuning PP 0.938272013	
RHIC Ring RHIC Ring ADO ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	apacMan (Web Version - Mo) Bookmarks Tools Help A A A A A A A A A A A A A A A A A A	Values Injection Tuning PP 0.938272013	
RHIC Ring ADO IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue IngSpec.blue	Bookmarks Tools Version - Mox Bookmarks Tools Help + A + Bookmarks Tools Help + A + Bookmarks Tools Help + Bookmarks Help +	Values Injection Tuning PP 0.938272013 1.0	
RHIC Status By speck ADO FingSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	Bookmarks Web Version - Max Bookmarks Book Help A	Values Injection Tuning PP 0.958272013 1.0 2.382677783808725	
Edit Miew History pecoma (Web Version G achines: 10000/pe RHIC Ring RHIC Status By speck ADO ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	Bookmarks Tools Version - Mox Bookmarks Tools Help + A + Bookmarks Tools Help + A + Bookmarks Tools Help + Bookmarks Help +	Values Injection Tuning PP 0.958272013 1.0 2.382677783808725 3.301167550506217	
RHIC Status By speck ADO FingSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	Bookmarks Web Version - Max Bookmarks Book Help A	Values Injection Tuning PP 0.958272013 1.0 2.382677783808725	
RHIC Status By speck ADO ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue ringSpec.blue	Beachmarks Web Version - Mox Bookmarks Book Help * \ 4 Sectors Bookmarks Book Help * \ 4 Bookmarks Help * \ 4 * \ 4 Bookmarks Help * \ 4 * \ 4 Bookmarks Help * \ 4 * \ 4	Values Injection Tuning PP 0.958272013 1.0 2.382677783808725 3.301167550506217	
Edit Miew History apecodan (Web Version Ge achlines: 0000/201 RHIC Status By special ADO FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue FingSpec.blue	Bookmarks Tools Version - Mox Bookmarks Tools Help A A	✓ ✓	
Edit View History Become and the Version Commentation (Web Version Commentation) Commentation RHIC Status By spect RHIC Status	ApacMan (Web Version - Max Bookmarks Tools Help A	Values Values Injection Tuning PP 0.958272013 1.0 2.382677783808725 3.391167550590217 19275 1440420305	
Edit Mew History Bedit Mew History Bedited Inno/April Bedi	Beachan (Web Version - Mox Bookmarks Book Jelp * \	Values Injection Tuning PP 0.938272013 1.0 2.382677783808725 3.301167550500217 19275 1440420305 480	

Figure 3: specMan: C++ (top) vs Vaadin (bottom)

 \sim 100Hz. Developers need to keep this in mind when deciding if Vaadin is good for the type of target applications which requires high frequency data updates.

The IO connections between front end application and Vaadin application server, and the IO connections between Vaadin application server and the rest of control system have to be properly controlled and coordinated, otherwise, the control system may be over burdened.

Some of the problems can be resolved by using proper system settings and program logic controls; Others may require software updates. For example, after upgrading from Glassfish 4.0 to v4.1, the UIDetachedException problem disappeared.

It is noticed that, with Vaadin, we can develop web applications without explicit knowledge of HTML and Javascript. However, since all client side web applications are essentially based on HTML, Javascript and CSS, familiarity with HTML, Javascript and CSS will help developers make web applications more flexible.

SUMMARY

The Vaadin web application frame work offers web application developers rich sets of UI components[3], add-ons, Java APIs, and powerful console level application like features. It makes accelerator control web application development simpler for Java developers without the prerequisite of HTML and Javascript The key aspects of accelerator control knowledge. applications include a fast and responsive bi-directional IO connection and UI interactions on web GUIs. The Vaadin web framework does a good job in this regard with server push features that support long polling or websockets. The testing of accelerator control web applications developed with Vaadin technology shows that, it is easy to convert console level control applications to HTML5 supported web platforms, and the web applications can be just as robust as the OS console level applications. As Vaadin technology and application server technology evolves, this web application framework will become more reliable. These technologies will aid in making web based accelerator control application development easy, powerful and more convenient to end users. The testing work with Vaadin showed some problems that needed attention during application development. Avoiding these pitfalls will help to develop robust and high performance web applications.

REFERENCES

- [1] Book of Vaadin: https://vaadin.com/book/
- [2] Wikipedia: https://en.wikipedia.org/wiki/Vaadin
- [3] Vaadin: http://demo.vaadin.com/sampler/