

# Browsium Ion Demo Script

---

This document provides a detailed demo script to enable you to effectively demonstrate Browsium Ion and highlight many of the key features. The demo includes showing an example (fictional) corporate line-of-business application working on a legacy Windows XP / IE6 PC, failing on a modern Windows 7 / IE9 PC, and then being fixed on the modern PC via a Browsium Ion configuration. The script includes comprehensive preparation instructions and demo steps to ensure the demo can be delivered in a consistent manner.

## System Preparation

In order to effectively demonstrate Browsium Ion, the following system configurations are required:

- 1) A PC or virtual machine running Windows XP, IE6 and Java 1.4
- 2) A PC or virtual machine running Windows 7, IE8 or IE9, Java 1.4, Java 7 (aka 1.7), the Ion Client Add-on and the Ion Configuration Manager

All systems should run the latest Windows Service Packs and be completely patched via Windows Update. However, it's critical to hide the IE8 update in Windows Update on Windows XP so that it remains running IE6. Without this step, Windows Update will automatically upgrade IE6 to IE8 and the demo will not function properly. A common demo configuration runs on a single PC with the native Windows 7 system used for configuration #2 and a virtual machine (with Windows XP Mode or Oracle VirtualBox) for configuration #1.

Java installations can be found on the Oracle website. It is recommended that you install the latest revision of each Java version. For Java 1.4, the latest (and last) revision is Java 1.4 update 19 ([link](#)). For Java 7, the latest revision as of the writing of this script is Java 7 update 7 ([link](#)), but new versions are released periodically. Install each version of Java to its own unique directory (which is the default installation behavior): for example install Java 1.4 to C:\program files (x86)\Java\j2re1.4.2\_19 and Java 7 to c:\program files (x86)\java\jre7. Note that newer versions of Java 7 will become available over time as Oracle services Java 7; in our experience the newer versions of Java 7 work fine for the demo. And be sure to install SE Runtime Environment and not the SE Development Kits! Their install web page might induce you to install the Development Kit accidentally.

## Demo Goals

At a high level, the goals of this script are to:



- Familiarize potential customers with the product & technology
- Demonstrate the ease with which Ion & web applications can be managed
- Demonstrate the negligibly-low end-user impact
- Demonstrate a variety of common scenarios that Ion can address (custom IE engines, custom Java engines and string replacement)
- Address the most common technical questions received from customers

## Additional Information

Additional resources are available to help you understand the flow of this demo. See this demo in action by watching the demo videos on our website at [www.browsium.com/ion/demos\\_and\\_resources/](http://www.browsium.com/ion/demos_and_resources/).

## Demo Script

Demo Steps	Context and Talking Points
<b>Section 1: The site working in Windows XP with IE6</b>	
Step 1.1: Open IE6 in Windows XP and navigate to a few sites	Demonstrate that you're running IE6. Point out the lack of tabs and IE6-era UI, open 'help about' from the Menu to show the IE6 version box. It's also a good idea to demonstrate how many popular Internet sites no longer support IE6. Youtube is a great example. Open <a href="http://www.youtube.com">www.youtube.com</a> in IE6 and point out how broken the experience is.
Step 1.2: Navigate to <a href="http://www.aggrid.com">www.aggrid.com</a>	Next show how a site that was designed during the IE6 era works very well in IE6 today. Navigate to <a href="http://www.aggrid.com">www.aggrid.com</a> . This is Browsium's demo site, used to show how various browsers handle the typical web pages used for enterprise line-of-business applications. Aggrid is a fictional company whose employees use the Aggrid.com portal to get company information and access corporate line-of-business applications. Point out that the demo will highlight functionality that works in the legacy environment (Windows XP, IE6, Java 1.4) but will not work in the modern environment (Windows 7, IE9, Java 6). The issues will include page layout problems, Java compatibility problems, and JavaScript compatibility problems.
Step 1.3: Demonstrate that the site is working as expected	Elements to highlight include: <ul style="list-style-type: none"> <li>a) The news ticker automatically advances through 5 stories.</li> <li>b) All links are aligned appropriately on the page.</li> <li>c) All section headers can be collapsed and expanded.</li> </ul>
Step 1.4: Expand the Alerts section header to the timesheet is overdue	Employees have been trained to use the Timesheets application under the HR menu to submit their timecards, but they occasionally need a reminder.
Step 1.5: Visit the 'Timesheets' page under the HR menu	When clicking the link on the menu, point out that the menu is the primary navigation tool for the site; most subpages are accessed through this menu. The menu opens automatically when you hover over the menu bar.
Step 1.6: Demonstrate the Timesheets application working properly (with Java 1.4)	The page will load the Timesheet application as a Java 1.4 object; demonstrate that you can enter hours in a cell for a given day (the cell will change color depending on the # of hours entered). Clicking the 'submit' button will do nothing; it's just a sample control, but right now you're just showing how it looks in IE6.
Step 1.7: Navigate to <a href="http://www.javatester.org">www.javatester.org</a>	Click the 'Test the version of Java...' link at the top of the page to show that Java 1.4 is being loaded in IE6 as it's the only version of Java installed on this PC. Point out that the timesheets application was written when Java 1.4 was the current version (about 10 years ago) and works well in that version. We'll see soon that it doesn't work in the current Java 6 releases.
Step 1.8: Navigate to Aggrid site and choose 'Dashboard' sub-menu item under the 'Finance' menu	This brings up the Finance Dashboard page. Click on the 'Customer Lookup' button at the bottom of the page. This will open a pop-up window.
Step 1.9: Type 'Acme Corp' into the customer lookup window and click Submit	A JavaScript call will retrieve a (fake) customer record. This is one more piece of the Aggrid app that will break under IE8 and IE9.  At this point you're done with IE6/Windows XP and you will do the rest of the demo in Win7/IE8 or IE9. Close IE6/XP
<b>Section 2: The site broken on Windows 7 with IE9</b>	
Step 2.1: Switch to Windows 7 with IE9	Demonstrate that you're in IE9 (or IE8); show the tabs and help.about box to prove which version of IE you're running. Remember to mention whether or not you're running the 64-bit version of Windows 7; Ion supports both 32-bit and 64-bit Windows 7 (but only the 32-bit version of Internet Explorer – the version that everyone uses).

	<p>Mention that the Ion client has already been installed on this PC but no rules or profiles have been configured. As such Ion, will sit harmlessly in the background, and all sites will be handled with IE's default setting (currently IE8 or IE9). Open the 'Manage add-ons' tool in IE9 and show the Ion client add-on in the list</p>
<p>Step 2.2: Navigate back to <a href="http://www.aggrid.com">www.aggrid.com</a> using IE9 and demonstrate the ways that the page is now broken</p>	<p>Elements to highlight include:</p> <ol style="list-style-type: none"> <li>An 'Upgrade to IE6' tab has appeared in the upper right-hand corner of the page (the developer who wrote this site did not anticipate an IE7, IE8, IE9 and IE10 following IE6)</li> <li>The menus are visible but unusable; the menus disappear when you move the mouse down to click on a submenu item</li> <li>All of the 'quick links' in the bottom right-hand corner are offset and overlapping other elements of the page</li> <li>None of the section headers will collapse or expand</li> <li>The news ticker is stuck and will not advance</li> </ol> <p>The menu &amp; quick links are victims of the modern IE layout engines and inconsistencies with how they deal with IE6-era design. The section headers &amp; news tickers are victims of the updated scripting engine (which has rightly been updated in IE7, IE8 and IE9 for performance, standards and security reasons).</p>
<p>Step 2.3: Try the site in Compatibility View: click the Compatibility View icon (the broken page icon) in the address bar so that it turns dark blue</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p>Off</p> </div> <div style="text-align: center;">  <p>On</p> </div> </div>	<p>Before tackling this site with Ion, let's first use the built-in tools Microsoft provides for fixing layout issues. So we'll turn on Compatibility View in IE9, which swaps in the IE7 layout engine, and see if this fixes our application.</p> <p>Some parts of the page are now fixed: the content is centered, the news ticker works, the quick links are lined up and the section headers can be collapsed, but several things are still broken:</p> <ul style="list-style-type: none"> <li>- The 'upgrade to IE6' tag still appears</li> <li>- Menus are still broken</li> </ul> <p>Compatibility View fixed some things but not everything; Compatibility View alone isn't enough to fix this site, so we'll use Ion to get it working. (Note that we'll need to go back after the demo and turn off Compatibility View so it's ready for the next demo session.)</p>
<p><b>Section 3: Get the site working with Ion</b></p>	
<p>Step 3.1: Close any open IE windows and open the Ion Configuration Manager</p>	<p>It's time to build an Ion configuration to get this site running properly using an older, but more compatible, IE engine and settings that are much closer to the settings used by IE6.</p> <p>It's important to note that the Ion Configuration Manager will only be used by the IT administrators within the organization responsible for getting the Ion client working with given web applications. Normal users/employees in a corporation will never see/use this tool and will not need to be trained in it. The Ion Configuration Manager is fully documented in our Administrator's Guide, which can be found at <a href="http://docs.browsium.com">http://docs.browsium.com</a> or within the Configuration Manager itself.</p>
<p>Step 3.2: Create a new Project for the demo's rules and profiles</p>	<p>From the File Menu, choose 'New Project' and give the project a meaningful name (Aggrid project). Note that as you adjust the name in the central pane of the Configuration Manager, the project name is updated in the object pane on the left.</p> <p>Once you've created your project, it's time to add a Profile</p>

<p>Step 3.3: Create a Profile for Aggrid</p>	<ol style="list-style-type: none"> <li>1) With your new project selected, click on 'Add Profile' either from the action pane on the right or from the File Menu on the left. Keep the default, which is an Adaptive IE Quirks engine profile.</li> </ol> <p>Each of these available Profiles is a different IE engine and browser settings that we can assign to an app or a site. Old IE6-dependent apps will typically run best under the 'Adaptive IE Quirks' engine, which runs the web page under either the Windows 2000-era IE Quirks IE engine or the IE7 engine, depending on how the app is written.</p> <ol style="list-style-type: none"> <li>2) Select 'OK' to create the profile and change the name to 'Aggrid with Adaptive Quirks (which is a little more descriptive than the generic name).</li> </ol> <p>Now the profile is ready to be used. There are many advanced options visible here that we can use to change the behavior of the IE engine selected, but for now we'll just try our site with the defaults to see how well or poorly it runs.</p>
<p>Step 3.4: Create a Rule to launch that Profile</p>	<ol style="list-style-type: none"> <li>1) With your new profile selected, choose 'Add Rule using this Profile' from the actions pane on the right.</li> <li>2) Paste the app's URL (<a href="http://aggrid.com">aggrid.com</a>) into the 'value' field.</li> </ol> <p>This rule says Absolute URI contains <a href="http://aggrid.com">aggrid.com</a> which basically says that this rule will fire when the entire URL as it's displayed in IE's address bar contains the string 'aggrid.com', but that's a little broad. Another domain might pass our domain name by reference and trigger the rule when we don't want it, so let's narrow the scope of the rule:</p> <ol style="list-style-type: none"> <li>3) Open the 'element' dropdown in the rules picker to show all the ways that a customer can customize a rule; choose 'Domain' from the list.</li> </ol> <p>Ion gives you tremendous flexibility to define Rules as narrowly or broadly as necessary for your application and environment. For purposes of our demo, we'll define the Rule to only trigger when it's an exact match on the domain.</p> <ol style="list-style-type: none"> <li>4) Open up the 'operator' dropdown in the rules picker to show all the ways we can customize the rule and choose 'Is' from the list.</li> </ol> <p>We offer many ways to write a rule, but for our demo we just need to fire our rule when the domain IS <a href="http://aggrid.com">aggrid.com</a>.</p> <ol style="list-style-type: none"> <li>5) Click 'OK' to save that rule. The Rules Manager is now shown in the Configuration Manager.</li> </ol> <p>Now you can see that we have one Rule in our list: when the DOMAIN IS <a href="http://aggrid.com">aggrid.com</a>, we'll load our Adaptive IE Quirks engine for that site instead of the default IE9 engine.</p>
<p>Step 3.5: Open the File Menu to discuss deployment options</p>	<p>Typically you'd deploy Rules &amp; Profiles with either Group Policy/Active Directory or a flat file. If you're using GP/AD, you can select the Active Directory option to export the settings, when you're done with them, to an AD-friendly format.</p>

	<p>If you have users who aren't managed by Policy (which may be an entire organization or just a subset of users, we allow a mixed environment with Ion), you can instead save those same settings as a flat file. Just choose 'Save Project As...' and save your settings out that way. It saves the same settings as the Policy option, just in a different format. However, because we're demoing this as one user on one PC, I don't need to deploy anything – I just need to save those settings locally with the 'Save Local Settings' option. Do so now.</p> <p>Keep the default Current User option from the resulting dialog. We have the option to commit these settings to just myself or every user account on this PC, but for the demo I'll keep it set just to me. Remind your audience that doing so will stop any running Ion processes, upload the new settings you've just specified (the new Rule &amp; Profile) and then restart the Ion processes. We recommend closing IE before Saving Local Settings to provide a cleaner demo. But this is not necessary – you can simply open a new IE tab after the Ion processes have restarted.</p> <p>Be sure to give the PC a few seconds to fully restart the Ion processes! Watch for the green Ion splash screen to appear and the close before restarting IE.</p>
Step 3.6: Restart IE9 and go back to <a href="http://www.aggrid.com">www.aggrid.com</a>	Mention that it doesn't matter how the end user navigates to a web site: It could be a link on the desktop or in email, or they could type it in the address bar or in the start menu – it doesn't matter, Ion will capture the navigation and instruct IE to open the website with the correct engine and settings regardless of how the user got there. In this way, the user just uses IE9 as they normally do, they don't have to do anything differently (or be trained in any way) and the right thing 'just happens.'
Step 3.7: Point out that the opened tab looks just like an IE9 tab	The UI for an end-user is identical whether they're running Ion or not. The tab behaves just like a regular IE9 tab, but Ion is managing the page – or more precisely, it's managing IE to ensure it uses the correct layout engine and settings to display the page. To prove it, right-click in white space on the page to show the Ion context menu. At the bottom of the menu it will say something like Ion 2.0.2 – Aggrid with Adaptive Quirks indicating the product (Ion), version # (2.0.2) and the Profile currently loaded (as you named it).
Step 3.8: Point out that the page is looking like it did under IE6	<p>The following elements are now working as they did back in IE6:</p> <ol style="list-style-type: none"> <li>1) The tag in the upper right that said Upgrade to IE6 is gone.</li> <li>2) The menus are now visible and usable (but don't click on them yet)</li> <li>3) The news ticker is advancing through the 5 stories.</li> <li>4) Section headers collapse and expand when clicked.</li> <li>5) The 'quick links' are placed correctly.</li> </ol>
Step 3.9: Open a 2 <sup>nd</sup> tab to <a href="http://www.youtube.com">www.youtube.com</a>	<p>Open a 2<sup>nd</sup> IE tab to a different site, one that's not covered by an Ion Rule. We'll use YouTube again to show how this modern site works great in a modern browser and can run side-by-side with a site managed by Ion using a legacy layout engine and settings.</p> <p>You can see that I now have two tabs open, side-by-side, the first running the site using the Adaptive IE Quirks engine and the second running the native IE9 engine. The user didn't have to do anything special to make that happen.</p>
Step 3.10: Summarize (before moving on to the Java example)	That shows how we easily solved the problem where a site was broken strictly because the wrong browser engine tried to render it.

	Next we'll show something more complicated: sometimes it's not as easy as just picking the right layout engine. Often we have to manage the ActiveX controls a site loads as well, and I'll show that now.
<b>Section 4: Solving the Java problem</b>	
Step 4.1: Click on the 'Timesheets' submenu under the 'HR' heading to bring up the Timesheets Java app	<p>You've already seen the Timesheets application running in IE6 with Java 1.4. But your Windows 7 PC has been upgraded to not only run IE9, but also the latest version of Java (Java 7), which was installed alongside Java 1.4 in the hope that we could use the new version of Java for new applications and secure Internet browsing and the old version of Java for our legacy applications that require it.</p> <p>Unfortunately Windows doesn't work that way. IE will always default to the most current version of Java unless the application specifically calls out an older version – and most applications (including our Timesheets application) do not. We can immediately see that something is wrong with this application under Java 7 – we just get one giant 'submit' button, which renders the application useless.</p> <p>You've upgraded to a modern OS and environment: Windows 7 + the latest version of Java. The Windows XP environment was your legacy environment: unsafe but compatible. This new environment is much more secure, but now has compatibility issues.</p>
Step 4.2: Open a 2 <sup>nd</sup> tab and navigate to <a href="http://javatester.org/version.html">javatester.org/version.html</a>	<p>Let's now confirm the version of Java that IE is invoking by visiting Javatester again. Remind your audience that this site has a Java application on it (the pink rectangle) that has one purpose: to report the version # of the Java runtime that's running it. It will report Java version: 1.7.0_## (## will vary depending on what update version of Java 7 you have installed).</p> <p>Since we haven't set up Ion to manage Java, that's the same version that the Adaptive IE Quirks engine in the other tab is running, and that's clearly broken. So let's tell Ion to load the older version of Java that we know worked for our site, too.</p>
Step 4.3: Close all open IE windows and switch back to the Ion Configuration Manager	As pointed out earlier, closing IE windows is not strictly required but it makes for a cleaner screen as you move between windows.
Step 4.4: Go back to your Adaptive IE Quirks profile in the manager	Point out the variety of options available to be toggled to get web sites/applications running. All of these options were added because we've found some applications that need them. They're all fully documented in our administrator's guide, but we don't need to change any of these other options to load the right version of Java for our Timesheets application.
Step 4.5: Select the 'Custom Files Manager' node under the profile	This is where we'll designate a specific version of Java to run. We'll create a new entry that will tell Ion to load the version of Java that's compatible with our Java applet when it's encountered, instead of the default version of Java that IE is currently loading.
Step 4.6: Click on the 'Add Custom File' link in the actions pane on the right	This works for any ActiveX control, like Flash or Silverlight as well, but for our demo we're using Java. Finding the correct files takes knowledge of Ion and the ActiveX control being used, but Browsium has already done the legwork for most of the popular ones and has documented these instructions on our Support site. We'll use those instructions to redirect Java for our Aggrid example.
Step 4.7: Click on the ellipsis button to the right of the 'Source' line (this will	For Java 1.4, navigate into your Java\j2re1.4.2_19\bin directory in Program Files (x86) and find NPJPI142_19.DLL. Select 'open' from the dialog to

open the file explorer)	populate the full path to the file: C:\Program Files (x86)\Java\j2re1.4.2_19\bin\NPJPI142_19.dll. This path will only work for 64-bit Windows 7. If you're running 32-bit Windows 7, omit '(x86)' and navigate to C:\Program Files\Java\j2re1.4.2_19\bin\NPJPI142_19.dll.
Step 4.8: Click on the ellipsis button to the right of the 'Target' line	For Java 7, navigate to your Java\jre7\bin directory in Program Files (x86) and find jp2iexp.dll. Select 'open' from the dialog to populate the full path to the file: C:\Program Files (x86)\Java\jre7\bin\jp2iexp.dll. Again, omit '(x86)' on 32-bit Windows 7. <i>[Note that for earlier updates of Java 7 (prior to Update 7), the required jp2iexp.dll is found at Oracle\Java FX 2.1 Runtime\bin directory. Try this path if the file in the jre7 directory doesn't work.]</i>
Step 4.9: Leave the 'register' checkbox Unchecked and click 'Ok' on the dialog	This will add one line to the list of custom files. Think of this as a file redirect: when Java content is encountered, Ion will load the version of Java you've specified in the left-hand column (version 1.4) instead of the version you've specified in the right-hand column (version 7). Note that for different versions of Java (1.5 and even between different versions of 6 or 7), different files may be specified. The full list of what files to pick based on what versions of Java you're running are detailed in a KB article up at the support site.
Step 4.10: Save those settings locally again	Let the audience know that there aren't any other settings to toggle and we don't have to update our Rule because it's already pointing to our Profile. So now all we have to do is save these settings again, give our processes a moment to restart, then we go back to our site and see how it looks.
Step 4.11: Launch IE and navigate back to <a href="http://www.aggrid.com">www.aggrid.com</a>	Note that the loading experience for the front page is the same despite our changes to the profile.
Step 4.12: Click on the 'timesheets' submenu under the HR heading	This will load the Java-enabled page again, only this time the page will load with Java 1.4. The loading experience is slightly slower with old Java, but after a few seconds, the grid control comes up the way we saw it work under IE6 and Java 1.4 originally. Enter some numbers into various days of the week to demonstrate it's working the same way as it was under IE6.
Step 4.13: Open a 2 <sup>nd</sup> tab and navigate back to <a href="http://javatester.org/version.html">javatester.org/version.html</a>	Demonstrate that this second tab is now open side-by-side with the first. In this second tab, you can see we're running IE9 & Java 7 because we have no rules set for this URL. But in the first tab, we have an old IE engine & Java 1.4.  This allows users to run the latest version of software (IE8/IE9 + Java 7) when on the open Internet but run the most compatible version of software (a legacy IE engine + Java 1.4) when needed for internal applications.
<b>Section 5: String replacement using Ion</b>	
Step 5.1: Any questions so far?	We're moving on to the last part of the demo: string replacement. This allows IT administrators to change web pages on the fly, but only for PCs running Ion. This means you can make adjustments to fix bugs and improve performance of web pages without touching the web server or affecting other client PCs.
Step 5.2: From <a href="http://www.aggrid.com/">http://www.aggrid.com/</a> , select 'Dashboard' from the 'Finance' menu	We're going to look at another type of issue you can easily address with Ion – incompatible JavaScript.  <ol style="list-style-type: none"> <li>1) Click the 'Customer Lookup' button at the bottom of the page. This will bring up a customer lookup pop-up.</li> <li>2) Type in 'Acme Corp' as we did in IE6 and hit the 'lookup' button – nothing happens.</li> </ol> Something about this part of the app is failing even though we're rendering the page in a legacy mode. Let's take a look at the code to see what the problem is.

<p>Step 5.3: Right-click on the pop-up window and choose 'View Source'</p>	<p>This shows the source for the page in the source viewer. We see the problem is line 8: notice that it says 'text\javascript' while the line above it says 'text/javascript' (the slash is going the other way). The problem is the '\'. IE6 was more permissive in which characters (and coding mistakes) it allowed, but IE9, even though it's rendering the page using the Quirks engine, refuses to run this improperly formatted code. This is because IE's JavaScript engine is separate from its layout engine, so our application is using IE9's much more modern and secure JavaScript engine which rejects this bad code.</p> <p>The fix is to replace the '\' with a '/' on that page. We could change the server, but that might have a ripple effect on client PCs we're not upgrading right now. Or we might not even have access to the server code if it's embedded in a proprietary application. Ion allows us to replace the code on the client just before it's processed by the browser – a very powerful capability.</p>
<p>Step 5.4: Copy the text in question: select text\javascript and copy it</p>	<p>Let's make a copy of the bad code so we can fix it in the Ion Profile.</p>
<p>Step 5.5: Switch to the Ion Configuration Manager and select the 'String Replacement Node' under the profile</p>	<p>Don't close any IE windows yet. We're going to instruct the Ion profile to update this broken JavaScript before loading the page with the Adaptive IE Quirks profile. The best part about this is that this fix only affects Ion clients running this Profile. All those other IE6 clients still out there are still being served the original HTML. They'll continue to work because fixing this with Ion doesn't touch the server.</p>
<p>Step 5.6: Choose 'Add String Replacement' from the action pane on the right</p>	<p>First let's paste in the bad text we want to fix.</p> <ol style="list-style-type: none"> <li>1) Paste in the text you copied in step 5.4 into the 'String to Find' textbox. This is the string that will be replaced.</li> <li>2) Paste the same string into the 'Replace with' text box. And this is the new text that we'll replace it with.</li> <li>3) Manually edit the '\' to '/' in the 'Replace with' box.</li> </ol> <p>String replacement is powerful but we don't want this taking effect on every page the user may visit with this profile. So we're going to limit this replacement to just the one page we're worried about.</p> <ol style="list-style-type: none"> <li>4) Switch back to the IE pop-up window and select and copy the entire URL 'http://www.aggrid.com/customer_lookup.html'</li> <li>5) Paste this into the 'Target URI' text box at the top of the string replacement window</li> </ol> <p>Leave the other options at their default (simple). Now we're done: we've specified the text we want to change, what it should change to, and have limited it to just the one page in question</p> <ol style="list-style-type: none"> <li>6) Click 'Ok' at the bottom of the dialog to commit the change to the profile.</li> <li>7) Close the IE pop-up window, the IE window and then 'Save Local Settings' in the Ion Configuration Manager.</li> </ol>
<p>Step 5.7: Test the change</p>	<ol style="list-style-type: none"> <li>1) Wait for the Ion settings to take effect.</li> <li>2) Open IE9 and navigate to <a href="http://www.aggrid.com">www.aggrid.com</a>.</li> <li>3) Navigate to the 'Dashboard' submenu under the 'Finance' menu header.</li> <li>4) Click on 'Customer lookup'.</li> <li>5) Type 'Acme Corp' into the text box and click on 'submit'.</li> </ol>



	<p>6) Note that the lookup now completes. The page is now working as expected.</p> <p>7) Right-click on the pop-up, view source, and note that line 8 now says 'text/javascript'. The page has been updated on the fly, and no other clients have been affected by this change.</p>
Step 5.8: Any questions?	This wraps up the demo of the core features of Ion. There's much more you can show, including the Feature Control Keys within Ion Profiles and the Ion Admin Toolbar which allows profiles to be swapped on the fly on a client system during testing.
<b>Section 6: Post-demo cleanup</b>	
Step 6.1: Clear Ion settings and reset Compatibility View to 'off' on the Windows 7 PC. No cleanup needed for Windows XP with IE6.	<p>This clears out your Rules &amp; Profiles from memory so that next time you run the demo, the critical elements of the page are appropriately broken again.</p> <ol style="list-style-type: none"> <li>1) Close all IE windows and select 'Clear Local Settings' from the File Menu of the Ion Configuration Manager on the Windows 7 system.</li> <li>2) Then launch IE again and navigate to <a href="http://www.aggrid.com">www.aggrid.com</a>. Click the Compatibility View icon (which should be dark blue) to turn it off (gray). Confirm that the Aggrid page layout is broken as it appeared in step 2.2.</li> </ol>