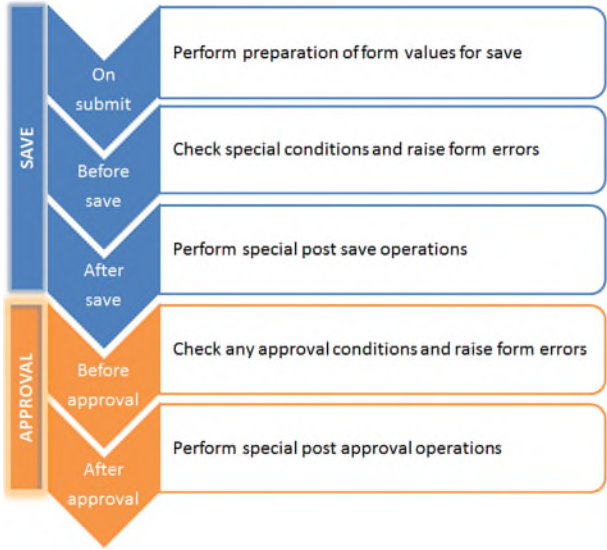
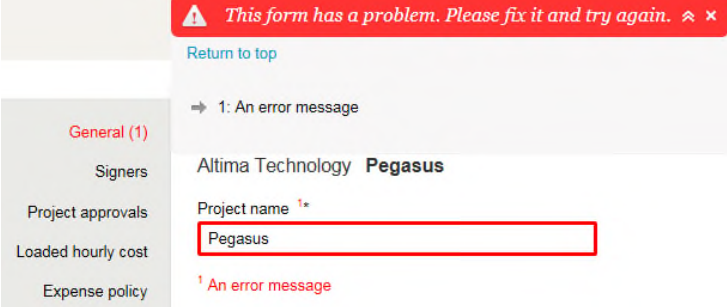


NSOA Functions	Events										
<table border="1"> <tr> <td data-bbox="99 163 410 380"> <b>context</b>                      + getAllParameters() : Array                      + getAllTerms() : Array                      + getParameter(name) : var                      + getTerm(terminid) : String                      + isTestMode() : Boolean                      + parseTerminology(message) : String                      + remainingTime() : Integer                      + remainingUnits() : Integer                 </td> <td data-bbox="410 163 773 275"> <b>meta</b>                      + alert(message) : Boolean                      + log(severity, message) : Boolean                      + sendMail(message) : Boolean                 </td> </tr> <tr> <td data-bbox="99 380 410 674"> <b>form</b>                      + confirmation(message) : Boolean                      + error(field, message) : Boolean                      + getAllValues() : Array                      + getLabel(field) : String                      + getName(field) : String                      + getNewRecord() : oaBase                      + getOldRecord() : oaBase                      + getValue(field) : var                      + get_value(field) : String                      + warning(message) : Boolean                      + setValue(field, value) : String                 </td> <td data-bbox="410 275 773 401"> <b>NSConnector</b>                      + integrateAllNow() : Boolean                      + integrateRecord() : Boolean                      + integrateWorkflowGroup(name) : Boolean                 </td> </tr> <tr> <td data-bbox="99 674 410 842"> <b>https</b>                      + delete(request) : Object                      + get(request) : Object                      + patch(request) : Object                      + post(request) : Object                      + put(request) : Object                 </td> <td data-bbox="410 401 773 485"> <b>record</b>                      + &lt;complexType&gt; { [id] } : oaBase                 </td> </tr> <tr> <td data-bbox="99 842 410 926"> <b>listview</b>                      + data(listviewId) : Iterator                      + list() : Array                 </td> <td data-bbox="410 485 773 590"> <b>report</b>                      + data(reportId, optionalParameters) : Iterator                      + list() : Array                 </td> </tr> <tr> <td data-bbox="99 926 410 961"></td> <td data-bbox="410 590 773 961"> <b>wsapi</b>                      + add(objects) : Array                      + approve(approveRequests) : Array                      + delete(objects) : Array                      + disableFilterSet(flag) : Boolean                      + enableLog(flag) : Boolean                      + modify(attributes, objects) : Array                      + read(readRequest) : Array                      + reject(rejectRequests) : Array                      + remainingTime() : Integer                      + submit(submitRequests) : Array                      + upsert(attributes, objects) : Array                      + unapprove(unapproveRequests) : Array                      + whoami() : oaUser                 </td> </tr> </table>	<b>context</b> + getAllParameters() : Array + getAllTerms() : Array + getParameter(name) : var + getTerm(terminid) : String + isTestMode() : Boolean + parseTerminology(message) : String + remainingTime() : Integer + remainingUnits() : Integer	<b>meta</b> + alert(message) : Boolean + log(severity, message) : Boolean + sendMail(message) : Boolean	<b>form</b> + confirmation(message) : Boolean + error(field, message) : Boolean + getAllValues() : Array + getLabel(field) : String + getName(field) : String + getNewRecord() : oaBase + getOldRecord() : oaBase + getValue(field) : var + get_value(field) : String + warning(message) : Boolean + setValue(field, value) : String	<b>NSConnector</b> + integrateAllNow() : Boolean + integrateRecord() : Boolean + integrateWorkflowGroup(name) : Boolean	<b>https</b> + delete(request) : Object + get(request) : Object + patch(request) : Object + post(request) : Object + put(request) : Object	<b>record</b> + <complexType> { [id] } : oaBase	<b>listview</b> + data(listviewId) : Iterator + list() : Array	<b>report</b> + data(reportId, optionalParameters) : Iterator + list() : Array		<b>wsapi</b> + add(objects) : Array + approve(approveRequests) : Array + delete(objects) : Array + disableFilterSet(flag) : Boolean + enableLog(flag) : Boolean + modify(attributes, objects) : Array + read(readRequest) : Array + reject(rejectRequests) : Array + remainingTime() : Integer + submit(submitRequests) : Array + upsert(attributes, objects) : Array + unapprove(unapproveRequests) : Array + whoami() : oaUser	
<b>context</b> + getAllParameters() : Array + getAllTerms() : Array + getParameter(name) : var + getTerm(terminid) : String + isTestMode() : Boolean + parseTerminology(message) : String + remainingTime() : Integer + remainingUnits() : Integer	<b>meta</b> + alert(message) : Boolean + log(severity, message) : Boolean + sendMail(message) : Boolean										
<b>form</b> + confirmation(message) : Boolean + error(field, message) : Boolean + getAllValues() : Array + getLabel(field) : String + getName(field) : String + getNewRecord() : oaBase + getOldRecord() : oaBase + getValue(field) : var + get_value(field) : String + warning(message) : Boolean + setValue(field, value) : String	<b>NSConnector</b> + integrateAllNow() : Boolean + integrateRecord() : Boolean + integrateWorkflowGroup(name) : Boolean										
<b>https</b> + delete(request) : Object + get(request) : Object + patch(request) : Object + post(request) : Object + put(request) : Object	<b>record</b> + <complexType> { [id] } : oaBase										
<b>listview</b> + data(listviewId) : Iterator + list() : Array	<b>report</b> + data(reportId, optionalParameters) : Iterator + list() : Array										
	<b>wsapi</b> + add(objects) : Array + approve(approveRequests) : Array + delete(objects) : Array + disableFilterSet(flag) : Boolean + enableLog(flag) : Boolean + modify(attributes, objects) : Array + read(readRequest) : Array + reject(rejectRequests) : Array + remainingTime() : Integer + submit(submitRequests) : Array + upsert(attributes, objects) : Array + unapprove(unapproveRequests) : Array + whoami() : oaUser										
<p><b>NSOA.form.error(field, errorMessage)</b></p> <pre>// displays 'An error message' next to the budget_time field function formError() {     NSOA.form.error("name","An error message"); }</pre> <ul style="list-style-type: none"> <li>Call this function to stop the form from saving</li> <li>Use this function to print an error message associated to the selected form field on the SuiteProjects Pro form.</li> <li>The first argument is the field name on the form where you want the message to show up. Pass an empty string to display the message at the top of the form.</li> <li>The message that appears looks exactly like the SuiteProjects Pro system generated errors.</li> </ul>	<p><b>Example</b></p> 										
<p><b>NSOA.form.getValue(field)</b></p> <pre>// writes the field value to the log function logFieldValue() {     NSOA.meta.alert('The date is ' + NSOA.form.get_value('date')); }</pre> <p><b>NSOA.form.getAllValues()</b></p> <pre>// get all the values on the fields on the form and write to log var allValues = NSOA.form.getAllValues(); for( var key in allValues ) {     NSOA.meta.alert(key + ' has value ' + allValues[key]); }</pre>	<p><b>Validation – Before save</b></p> <pre>// compare two date fields on a receipt // NOTE: call on 'Before save' form event function validateTravelDates() {     var receiptDate = NSOA.form.getValue('date');     var travelDate = NSOA.form.getValue('TravelDate_c');     if ( receiptDate &lt; travelDate ) {         NSOA.form.error('TravelDate_c',             'The travel date cannot be after the receipt date!');     } }</pre>										
<p><b>NSOA.meta.log(type,message)</b></p> <pre>// writes an error severity message to the log function logMessage() {     NSOA.meta.log('error', 'Travel date is after receipt date'); }</pre> <p><b>NSOA.meta.alert(message)</b></p> <pre>// writes an info severity message to the log function logMessage() {     NSOA.meta.alert('Travel date is after receipt date'); }</pre>	<p><b>Log</b></p> <p>Script Deployment Messages</p> <table border="1"> <thead> <tr> <th>Severity</th> <th>Timestamp</th> <th>Generated by</th> <th>Message</th> <th>User</th> </tr> </thead> <tbody> <tr> <td>Info</td> <td>2013-08-12 07:0</td> <td>System</td> <td>NSOA.form.getLabel2 is not a function</td> <td>Collins</td> </tr> </tbody> </table> <p>1 row</p> <p><b>Severity</b> Fatal, Error, Warning, Info, Debug, or Trace  <b>Timestamp</b> time the message was logged  <b>Generated by</b> your script or the system  <b>Message</b> full message text  <b>User</b> who was running the script</p>	Severity	Timestamp	Generated by	Message	User	Info	2013-08-12 07:0	System	NSOA.form.getLabel2 is not a function	Collins
Severity	Timestamp	Generated by	Message	User							
Info	2013-08-12 07:0	System	NSOA.form.getLabel2 is not a function	Collins							

<b>NSOA.record.&lt;complex type&gt;( [id] )</b> <pre>function testCreatingObjects {     // create category object populated with data for id = 5     var populatedCategory = NSOA.record.oaCategory(5);     // create a blank category object     var blankCategory = NSOA.record.oaCategory(); }</pre>		<b>Creating Records</b> <pre>// create a new category function createCategory {     // create an empty category object     var category = new NSOA.record.oaCategory();     // fill out the details – the ID is created by SuiteProjects Pro     category.name = "New Category";     category.cost_centerid = "123";     category.currency = "USD";     // invoke the add call     var results = NSOA.wsapi.add( [category] );     // log the ID of the new category     NSOA.meta.alert("The category id is " + results[0].id); }</pre>																	
<b>NSOA.wsapi.add(objects)</b> <ul style="list-style-type: none"> <li>See the example on the right</li> <li>The function takes an array of objects</li> <li>[category] // This is an array with one category object</li> <li>You can pass up to 1000 objects</li> <li>The ID is created by the system (returned in the result)</li> </ul>		<b>Modifying Records</b> <pre>// This is called on the "After save" event for the Project form function append_to_project_notes() {     // get the saved project values     var newr = NSOA.form.getNewRecord();     // Invoke the add call     var project = new NSOA.record.oaProject();     project.id = newr.id; // use the saved project ID     project.notes = newr.notes + "\nAppended: " + (new Date()).toString();     // disable user filters to make sure it runs for all users     NSOA.wsapi.disableFilterSet(true);     // invoke the modify call     var arrayOfupdateResult = NSOA.wsapi.modify([], [project]);     // log the modify status     NSOA.meta.alert("Got modify status: " + arrayOfupdateResult[0].status); }</pre>																	
<b>NSOA.form.getNewRecord()</b> <ul style="list-style-type: none"> <li>See the example on the right</li> <li>Gets the entity record for a form with the newly saved values</li> <li>This function should be called on the <b>After save event</b></li> <li>The function returns <b>null</b> if called before the form has been saved</li> <li><b>NSOA.form.getOldRecord()</b> returns the current (unsaved) values</li> </ul>		<b>NSOA.wsapi.disableFilterSet( [flag] )</b> <pre>function testDisableFilterSet {     // disables user filter sets on .wsapi requests     NSOA.wsapi.disableFilterSet(true);     // enables user filterset on .wsapi requests (default)     NSOA.wsapi.disableFilterSet(false);     // returns the Boolean state (without changing setting)     var filtersDisabled = NSOA.wsapi.disableFilterSet(); } // Hint: set NSOA.wsapi.enableLog(true) to show SOAP messages</pre>																	
<b>Scripting Governance</b> <p>To prevent scripts from consuming excessive resources or running out of control, two limitations are placed on scripts:</p> <p><b>Time Limit</b> — If a form script runs for more than 5 seconds (not including wsapi call time) it is automatically terminated with a form error.</p> <p>If a request (all scripts triggered by the same form save) uses more than 60 seconds of wsapi time it will also be automatically terminated.</p> <p><i>Scheduled scripts are allowed 1 hour of JS runtime and 1 hour of wsapi.</i></p> <p><b>Units Limit</b> — NSOA functions are assigned a unit value. Each time an NSOA function is called its unit value is consumed. A form script is allowed to consume a maximum of 1000 units with each run before it is automatically terminated with a form error.</p> <p><i>Scheduled scripts are allowed 1, 000, 000 units.</i></p> <p><b>SendMail Limit</b> — Forms scripts are allowed to send a maximum of 3 emails on a single event.</p> <p><i>Scheduled scripts are allowed to send a maximum of 100 emails on a single scheduled event.</i></p>		<b>Governance Units</b> <table border="1"> <tr> <td>NSOA.context.getTerm(termid) NSOA.context.isTestMode() NSOA.context.remainingTime() NSOA.context.remainingUnits() NSOA.record.&lt;complex type&gt;( [id] ) NSOA.wsapi.remainingTime()</td> <td>0</td> </tr> <tr> <td>NSOA.context.getParameter(name) NSOA.form.confirmation(message) NSOA.form.error(field, message) NSOA.form.getLabel(field) NSOA.form.getName(field) NSOA.form.getNewRecord() NSOA.form.getOldRecord() NSOA.form.getValue(field) NSOA.form.get_value(field) NSOA.form.warning(message) NSOA.listview.list(), NSOA.report.list() NSOA.wsapi.disableFilterSet( [ flag ] ) NSOA.wsapi.enableLog( [ flag ] ) NSOA.wsapi.whoami()</td> <td>1</td> </tr> <tr> <td>NSOA.context.parseTerminology(message) NSOA.meta.alert(message) NSOA.meta.log(severity, message)</td> <td>4</td> </tr> <tr> <td>NSOA.context.getAllParameters() NSOA.context.getAllTerms() NSOA.form.getAllValues() NSOA.https.delete(request) NSOA.https.get(request) NSOA.https.patch(request) NSOA.https.post(request) NSOA.https.put(request) NSOA.meta.sendMail(msg) NSOA.NSConnector.integrateRecord()</td> <td>10</td> </tr> <tr> <td>NSOA.listview.data(listviewId) NSOA.report.data(reportId, optionalParameters)</td> <td>10 for each page loaded into iterator on demand (max 1000 items per page)</td> </tr> <tr> <td>NSOA.wsapi.read(readRequest) NSOA.wsapi.add(objects) NSOA.wsapi.delete(objects)</td> <td>20 (+10 per additional object)</td> </tr> <tr> <td>NSOA.wsapi.modify(attributes, objects) NSOA.wsapi.upsert(attributes,objects)</td> <td>40 (+20 per additional object)</td> </tr> <tr> <td>NSOA.NSConnector.integrateAllNow() NSOA.NSConnector.integrateWorkflowGroup(name)</td> <td>1000</td> </tr> </table>		NSOA.context.getTerm(termid) NSOA.context.isTestMode() NSOA.context.remainingTime() NSOA.context.remainingUnits() NSOA.record.<complex type>( [id] ) NSOA.wsapi.remainingTime()	0	NSOA.context.getParameter(name) NSOA.form.confirmation(message) NSOA.form.error(field, message) NSOA.form.getLabel(field) NSOA.form.getName(field) NSOA.form.getNewRecord() NSOA.form.getOldRecord() NSOA.form.getValue(field) NSOA.form.get_value(field) NSOA.form.warning(message) NSOA.listview.list(), NSOA.report.list() NSOA.wsapi.disableFilterSet( [ flag ] ) NSOA.wsapi.enableLog( [ flag ] ) NSOA.wsapi.whoami()	1	NSOA.context.parseTerminology(message) NSOA.meta.alert(message) NSOA.meta.log(severity, message)	4	NSOA.context.getAllParameters() NSOA.context.getAllTerms() NSOA.form.getAllValues() NSOA.https.delete(request) NSOA.https.get(request) NSOA.https.patch(request) NSOA.https.post(request) NSOA.https.put(request) NSOA.meta.sendMail(msg) NSOA.NSConnector.integrateRecord()	10	NSOA.listview.data(listviewId) NSOA.report.data(reportId, optionalParameters)	10 for each page loaded into iterator on demand (max 1000 items per page)	NSOA.wsapi.read(readRequest) NSOA.wsapi.add(objects) NSOA.wsapi.delete(objects)	20 (+10 per additional object)	NSOA.wsapi.modify(attributes, objects) NSOA.wsapi.upsert(attributes,objects)	40 (+20 per additional object)	NSOA.NSConnector.integrateAllNow() NSOA.NSConnector.integrateWorkflowGroup(name)	1000
NSOA.context.getTerm(termid) NSOA.context.isTestMode() NSOA.context.remainingTime() NSOA.context.remainingUnits() NSOA.record.<complex type>( [id] ) NSOA.wsapi.remainingTime()	0																		
NSOA.context.getParameter(name) NSOA.form.confirmation(message) NSOA.form.error(field, message) NSOA.form.getLabel(field) NSOA.form.getName(field) NSOA.form.getNewRecord() NSOA.form.getOldRecord() NSOA.form.getValue(field) NSOA.form.get_value(field) NSOA.form.warning(message) NSOA.listview.list(), NSOA.report.list() NSOA.wsapi.disableFilterSet( [ flag ] ) NSOA.wsapi.enableLog( [ flag ] ) NSOA.wsapi.whoami()	1																		
NSOA.context.parseTerminology(message) NSOA.meta.alert(message) NSOA.meta.log(severity, message)	4																		
NSOA.context.getAllParameters() NSOA.context.getAllTerms() NSOA.form.getAllValues() NSOA.https.delete(request) NSOA.https.get(request) NSOA.https.patch(request) NSOA.https.post(request) NSOA.https.put(request) NSOA.meta.sendMail(msg) NSOA.NSConnector.integrateRecord()	10																		
NSOA.listview.data(listviewId) NSOA.report.data(reportId, optionalParameters)	10 for each page loaded into iterator on demand (max 1000 items per page)																		
NSOA.wsapi.read(readRequest) NSOA.wsapi.add(objects) NSOA.wsapi.delete(objects)	20 (+10 per additional object)																		
NSOA.wsapi.modify(attributes, objects) NSOA.wsapi.upsert(attributes,objects)	40 (+20 per additional object)																		
NSOA.NSConnector.integrateAllNow() NSOA.NSConnector.integrateWorkflowGroup(name)	1000																		
<b>NSOA.context.remainingUnits()</b> <pre>// displays remaining units as a form error message function displayRemainingUnits () {     NSOA.form.error("", "Units: " + NSOA.context.remainingUnits()); }</pre>		<b>Scripting Documentation</b> <a href="#">User Scripting</a> <a href="#">XML API &amp; SOAP API</a> <a href="#">NetSuite Integration</a> <a href="#">Business Intelligence Connector</a>																	
<b>Editor Shortcuts</b> <table border="1"> <tr> <td>&lt;Ctrl&gt; + &lt;SPACE&gt;</td> <td>Show Auto List (after "NSOA.")</td> </tr> <tr> <td>&lt;Ctrl&gt; + &lt;F11&gt;</td> <td>Full-screen editor (&lt;Esc&gt; to exit)</td> </tr> <tr> <td>&lt;Ctrl&gt; + F</td> <td>Start a search</td> </tr> <tr> <td>&lt;Ctrl&gt; + G</td> <td>Find next</td> </tr> <tr> <td>&lt;Shift&gt; + &lt;Ctrl&gt; + G</td> <td>Find previous</td> </tr> <tr> <td>&lt;Shift&gt; + &lt;Ctrl&gt; + F</td> <td>Replace</td> </tr> <tr> <td>&lt;Shift&gt; + &lt;Ctrl&gt; + R</td> <td>Replace all</td> </tr> <tr> <td>&lt;Alt&gt; + G</td> <td>Activate Jump to Line dialog</td> </tr> </table>		<Ctrl> + <SPACE>	Show Auto List (after "NSOA.")	<Ctrl> + <F11>	Full-screen editor (<Esc> to exit)	<Ctrl> + F	Start a search	<Ctrl> + G	Find next	<Shift> + <Ctrl> + G	Find previous	<Shift> + <Ctrl> + F	Replace	<Shift> + <Ctrl> + R	Replace all	<Alt> + G	Activate Jump to Line dialog		
<Ctrl> + <SPACE>	Show Auto List (after "NSOA.")																		
<Ctrl> + <F11>	Full-screen editor (<Esc> to exit)																		
<Ctrl> + F	Start a search																		
<Ctrl> + G	Find next																		
<Shift> + <Ctrl> + G	Find previous																		
<Shift> + <Ctrl> + F	Replace																		
<Shift> + <Ctrl> + R	Replace all																		
<Alt> + G	Activate Jump to Line dialog																		