

# Client-Side Integration with Visualforce Pages

*Interacting with Visualforce via JavaScript*

*Nathan Lipke, salesforce.com, Lead Member of Technical Staff, @evilN8*

*Mike Tetlow, Bracket Labs, Lead Developer, @Mikename*



SELL. SERVICE. MARKET. SUCCEED.



# Safe harbor

Safe harbor statement under the Private Securities Litigation Reform Act of 1995:

This presentation may contain forward-looking statements that involve risks, uncertainties, and assumptions. If any such uncertainties materialize or if any of the assumptions proves incorrect, the results of salesforce.com, inc. could differ materially from the results expressed or implied by the forward-looking statements we make. All statements other than statements of historical fact could be deemed forward-looking, including any projections of product or service availability, subscriber growth, earnings, revenues, or other financial items and any statements regarding strategies or plans of management for future operations, statements of belief, any statements concerning new, planned, or upgraded services or technology developments and customer contracts or use of our services.

The risks and uncertainties referred to above include – but are not limited to – risks associated with developing and delivering new functionality for our service, new products and services, our new business model, our past operating losses, possible fluctuations in our operating results and rate of growth, interruptions or delays in our Web hosting, breach of our security measures, the outcome of any litigation, risks associated with completed and any possible mergers and acquisitions, the immature market in which we operate, our relatively limited operating history, our ability to expand, retain, and motivate our employees and manage our growth, new releases of our service and successful customer deployment, our limited history reselling non-salesforce.com products, and utilization and selling to larger enterprise customers. Further information on potential factors that could affect the financial results of salesforce.com, inc. is included in our annual report on Form 10-K for the most recent fiscal year and in our quarterly report on Form 10-Q for the most recent fiscal quarter. These documents and others containing important disclosures are available on the SEC Filings section of the Investor Information section of our Web site.

Any unreleased services or features referenced in this or other presentations, press releases or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make the purchase decisions based upon features that are currently available. Salesforce.com, inc. assumes no obligation and does not intend to update these forward-looking statements.





**Nathan Lipke**

*Lead Member of Technical Staff*  
**@evilN8**

[bracket] labs

**Mike Tetlow**  
*Lead Developer*  
*@Mikename*

# Bracket Labs and TaskRay Project Management

TaskRay  
Sales & Marketing

Add a task and hit enter to Quick Add [New Task](#)

Column Deadline Plan

Holding 2

- Book venue for Dreamforce party  
Dreamforce  
Nov 22

Prioritized 2

- Salesforce MVP Webinar Implementations  
Nov 21
- Call Daryl Shaber  
Dreamforce  
Sep 16

Started 3

- Review contract with Sales Manager  
Sales Fulfillment  
Oct 18
- Finalize Copy  
Advertising  
Oct 24
- Reserve booth  
Dreamforce  
Dec 01

CHATTER GROUP [i](#)  
Sales & Marketing

CONTRIBUTORS [i](#)  
Group Members

PROJECTS

Select All / Deselect All

- Advertising
- Dreamforce
- Implementations
- Sales Fulfillment

The screenshot displays the TaskRay project management interface. At the top, there are navigation icons for Column, Deadline, and Plan, and links for Preferences and Feedback. Below the header, there are tabs for Holding, Prioritized, Started, and Finished, with Started being the active tab. The main area shows three columns of tasks:

- Holding:** 2 tasks. One for "Book venue for Dreamforce party" due Nov 22.
- Prioritized:** 2 tasks. One for "Salesforce MVP Webinar Implementations" due Nov 21, and one for "Call Daryl Shaber" due Sep 16.
- Started:** 3 tasks. One for "Review contract with Sales Manager" due Oct 18, one for "Finalize Copy" due Oct 24, and one for "Reserve booth" due Dec 01.

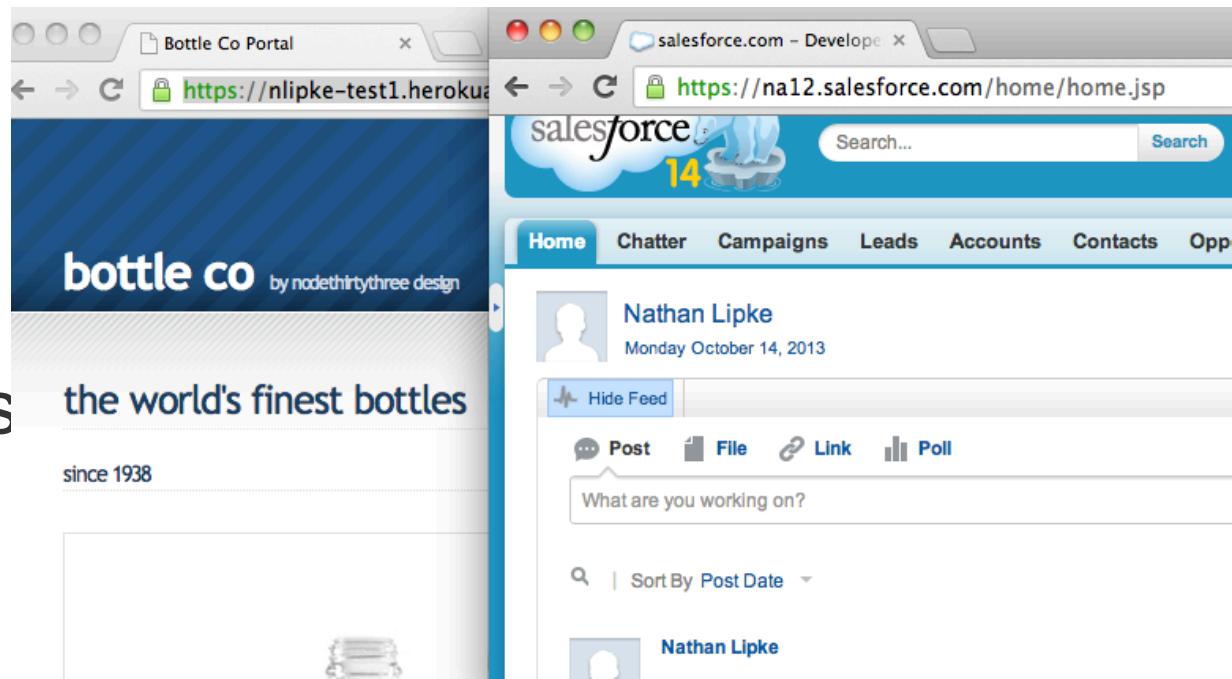
On the right side, there are sections for Chatter Group (Sales & Marketing), Contributors (Group Members), and Projects (Advertising, Dreamforce, Implementations, Sales Fulfillment). A legend at the bottom identifies the colors for each project category. The interface uses a clean, modern design with a light blue header and white background.



# You Intranet/Portal

Just about every organization has their own intranet, portal or wiki in addition to their Salesforce organization.

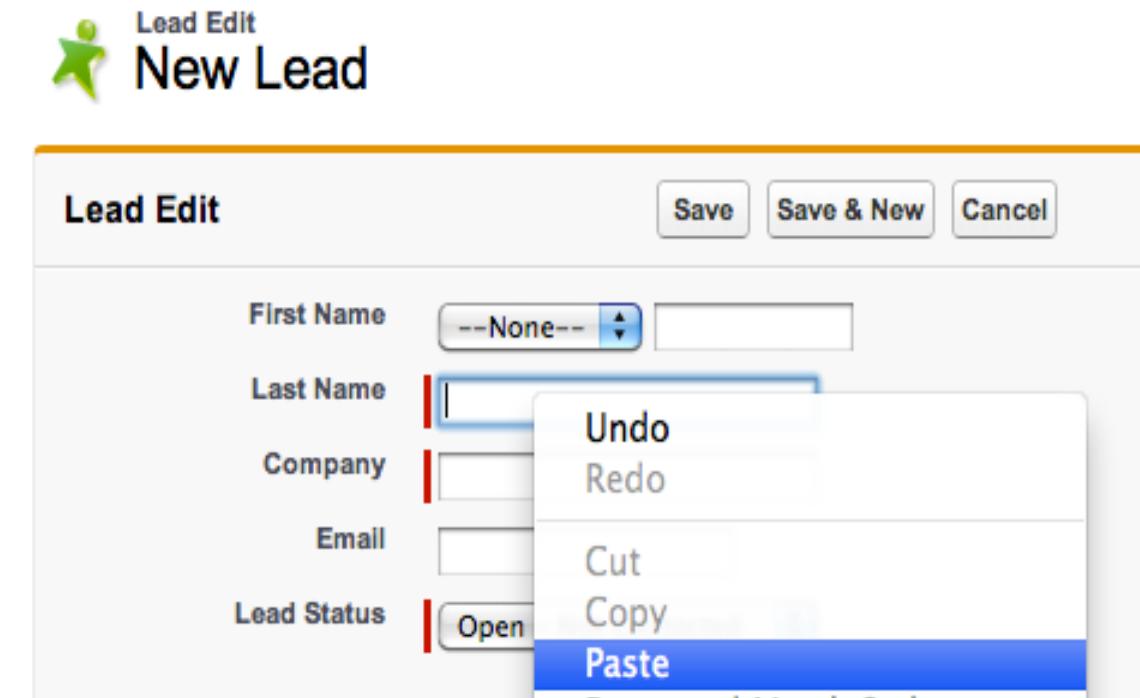
- Typically these are separate applications which do not interact with Salesforce.
- However, they often contain applications which contain data which is used in Salesforce.



# Unintegrated pages cause pain

Unintegrated pages provide poor usability:

- Copy and pasting
  - Painful
  - Error prone
    - Wrong or missed fields
    - Typos
- Refreshing pages is a poor solution
  - May be forgotten, causing out-of-date info to be displayed



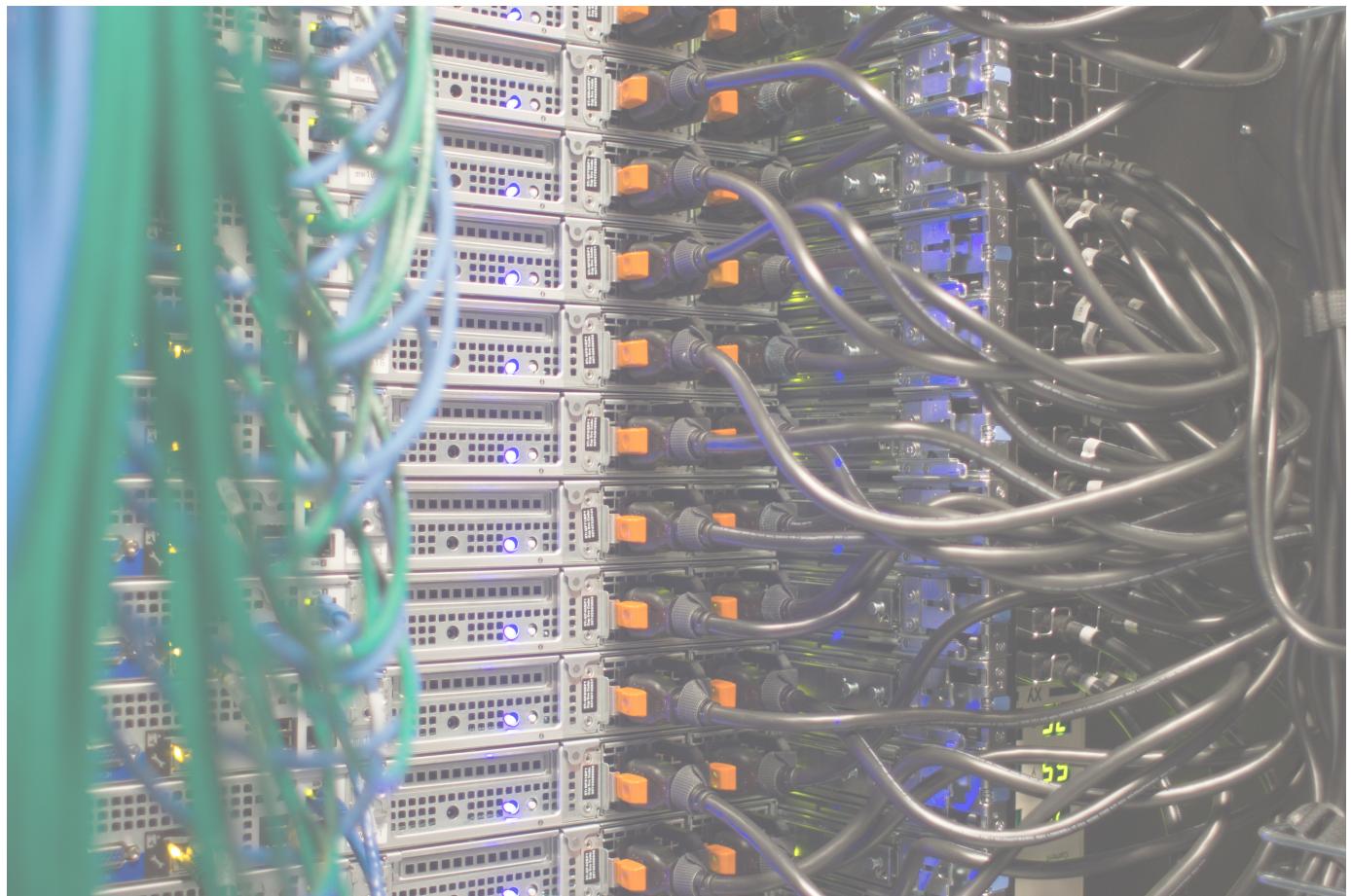
# Integration is traditionally server-side

Currently integration with Salesforce currently written on a server:

- Java, .NET, Ruby, ...

Which connect to Salesforce via:

- REST API
- SOAP API
- Bulk
- Meta-Data
- Chatter Connect



# Interactive Web Pages

Rich web experience is no longer optional. Users expect a rich, interactive experience on web-sites.

- However, this breaks down when the user needs to use different pages
  - Each page is it's own silo
- Interactions between pages offers:
  - Seamless experience
  - Ease of use
  - Single view of data



# Client-Side Integration

The world is moving to client-side programming:

- HTML5/JavaScript is the Web Standard programming language
- More interactive/responsive
- Less hops
  - No need server to proxy between the client and salesforce.com
- JavaScript integrates with Apex
  - Action Functions
  - Remote Actions
  - REST APIs/Resources can be called from JavaScript



# Client-Side Integration

How to make your page interact with a Visualforce page:

1. Open a new window in JavaScript

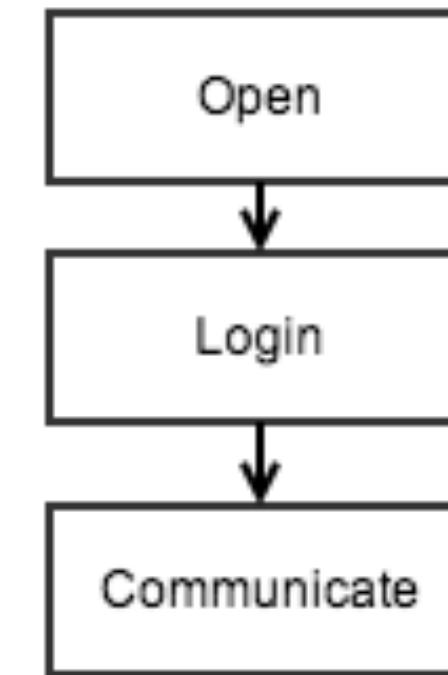
- `window.open()`

2. Login:

- Standard Salesforce login
- SSO (SAML)
- Oauth

3. Communicate

- Web Messaging
- Drag and drop



# Open a new window

In your parent window call JavaScript to open VF Page.

- `open()` Must be in an onclick handler to avoid popup-blockers.
- Save the return for sending message

```
childWin = window.open(  
    'https://c.na15.visual.force.com/apex/trprojectboard',  
    'childWin',  
    'status=0,toolbar=0,menubar=0,resizable=0,scrollbars=0,top=400,left=0,  
    height=300,width=1280')
```



# Login

- Standard login
  - User enters username and password
  - No coding, no configuration, no server-side integration
- SSO (SAML)
  - No username/password required
  - No coding, configuration work needed
  - Server-side integration not supported
- OAuth
  - Username/password and acceptance required\*
  - Coding and configuration work needed
  - Use for server-side integration



# Security: Salesforce

- Standard Salesforce security works:
  - Users cannot do anything they can't normally do
    - Profiles, Perm Sets, Sharing, Ownership, ...
  - Use **with sharing** in your Apex classes
- SAML
  - Is the same user logged into your intranet site
- Oauth
  - Limit Scope
  - Use with remote action calls and REST calls
  - Revocable



# Communicating

- Web Messaging

- Allows JavaScript messaging between windows
- Works regardless of domains
- Need a handle to the window
  - External page: return from `window.open()`
  - Visualforce: Use `window.opener`

- Drag and Drop

- Must come from an user gesture
- Drag page: sets the data
- Drop page: get the data



# Code: Sending a message (External Page)

```
function createTask(name) {  
    childWin.postMessage( { //message  
        name : 'createTask',  
        payload : {  
            Name : 'New Task',  
            TASKRAY__project__c : 'a02G0000006WoYw'  
        }  
    },  
    'https://c.na15.visual.force.com' //target origin  
);  
}
```



# Code: Receiving messages (Visualforce)

```
//Start listening  
window.addEventListener("message", receiveMessage, false);  
  
function receiveMessage(event) {  
    if (event.origin !== 'https://nlipke-test1.herokuapp.com')  
        return; //ignore it  
    if (event.data.name === 'newTask')  
        newTask(event.data.payload.name, event.data.payload.project);  
}
```



# Security: Messaging

- Know who is messaging you and who you're messaging
  - Use My Domain
  - Check event.origin in handlers
  - Don't use '\*' for the origin in postMessage()
- Validate messages
  - Validate all fields
  - Signed Messages
- Warning: Messages can be sent from Developer Tools!
- Use server-side integration as needed



# Code: Drag and Drop (External Page)

```
el.setAttribute('draggable', 'true');  
el.addEventListener('dragstart', function (e) {  
    e.dataTransfer.effectAllowed = 'all';  
    e.dataTransfer.setData('Task', JSON.stringify(getTask()));  
});
```



# Code: Drag and Drop (Visualforce)

```
el.addEventListener('drop', function(e) {  
    if (e.stopPropagation) e.stopPropagation();  
    createTask(JSON.parse(e.dataTransfer.getData('Task')));  
    el.style.background="gray";  
});  
  
el.addEventListener('dragover', function(e) {  
    if (e.preventDefault) e.preventDefault(); // allows us to drop  
    el.style.background="red";  
});
```



# Code: Drag and Drop (Visualforce cont.)

```
el.addEventListener('dragenter', function(e) {  
    el.style.background="red";  
}) ;  
  
el.addEventListener('dragleave', function(e) {  
    el.style.background="gray";  
}) ;
```



# Security: Drag and Drop

- Drag and drop does not provide an origin
  - Can be from any browser window
    - Even a different browser (e.g. Firefox->Chrome)
  - Can be from a non-browser application
    - File system
    - Other apps
- Verify drop data via messaging
  - Proves the origin



# Calling your Apex Controller

There are two ways to call Apex from JavaScript

- <apex:actionFunction> tag
  - No parameters
  - Allows rerender targets
  - Can call non-static controller methods
- Remote Actions
  - Parameters
  - Only static controller methods
  - Has asynchronous JavaScript callback (may call ActionFunction to rerender)



# Code: Action Function (Visualforce)

Apex:

```
public PageReference draw() {  
    recalculateData();  
    return null;  
}
```

Visualforce:

```
<apex:actionFunction name="draw" action="{!draw}" rerender="node"/>
```

JavaScript:

```
draw();
```



# Code: Remote Action Apex (Visualforce)

Apex:

```
public class MyController with sharing {  
    @RemoteAction  
    public static String newTask(TASKRAY__Project_Task__c task) {  
        insert task;  
        return task.id;  
    }  
}
```



# Code: Remote Action JavaScript (Visualforce)

JavaScript:

```
MyController.createTask (  
    {Name : name, TASKRAY__Project__c : projectId},  
    function (result, event) {  
        if (event.status) {  
            taskId = result;  
            draw(); //Call the Action Function  
        }  
    } );
```



# Demo



# Requirements

- Browser
  - Something written this decade
  - IE 8+
  - Firefox
  - Chrome
  - Safari
- Perms
  - Must be able to author Visualforce and Apex (Author Apex perm)



# Q&A



Nathan Lipke

*Lead Member of Technical Staff,  
@evilN8*



Mike Tetlow

*Lead Developer,  
@Mikename*



# Links

- Window.open: <https://developer.mozilla.org/en-US/docs/Web/API/window.open>
- Web Messaging: <https://developer.mozilla.org/en-US/docs/Web/API/window.postMessage>
- Drag and Drop: [https://developer.mozilla.org/en-US/docs/DragDrop/Drag\\_and\\_Drop](https://developer.mozilla.org/en-US/docs/DragDrop/Drag_and_Drop)



# Links

- SSO (SAML): [http://help.salesforce.com/help/pdfs/en/salesforce\\_single\\_sign\\_on.pdf](http://help.salesforce.com/help/pdfs/en/salesforce_single_sign_on.pdf)
- Oauth: [https://help.salesforce.com/apex/HTViewHelpDoc?id=remoteaccess\\_authenticate.htm&language=en](https://help.salesforce.com/apex/HTViewHelpDoc?id=remoteaccess_authenticate.htm&language=en)



# Links

- Remote Actions: [http://www.salesforce.com/us/developer/docs/pages/Content/pages\\_js\\_remoting.htm](http://www.salesforce.com/us/developer/docs/pages/Content/pages_js_remoting.htm)
- ActionFunction: [http://www.salesforce.com/us/developer/docs/pages/Content/pages\\_compref\\_actionFunction.htm](http://www.salesforce.com/us/developer/docs/pages/Content/pages_compref_actionFunction.htm)



# Thank you.

