Lightning Components

Denver Technology Day

Troy Hedges
Principal Solution Engineer
thedges@salesforce.com

Forward Looking Statements

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.

Agenda

Lightning Components

- ❑ Lightning Experience Overview
- ❑ Lightning Component Overview
- ❑ Extras
- ❑ Resources
- ❑ Demo

Lightning Experience Overview

Introducing Lightning

A new way to go faster with Salesforce

Lightning Experience

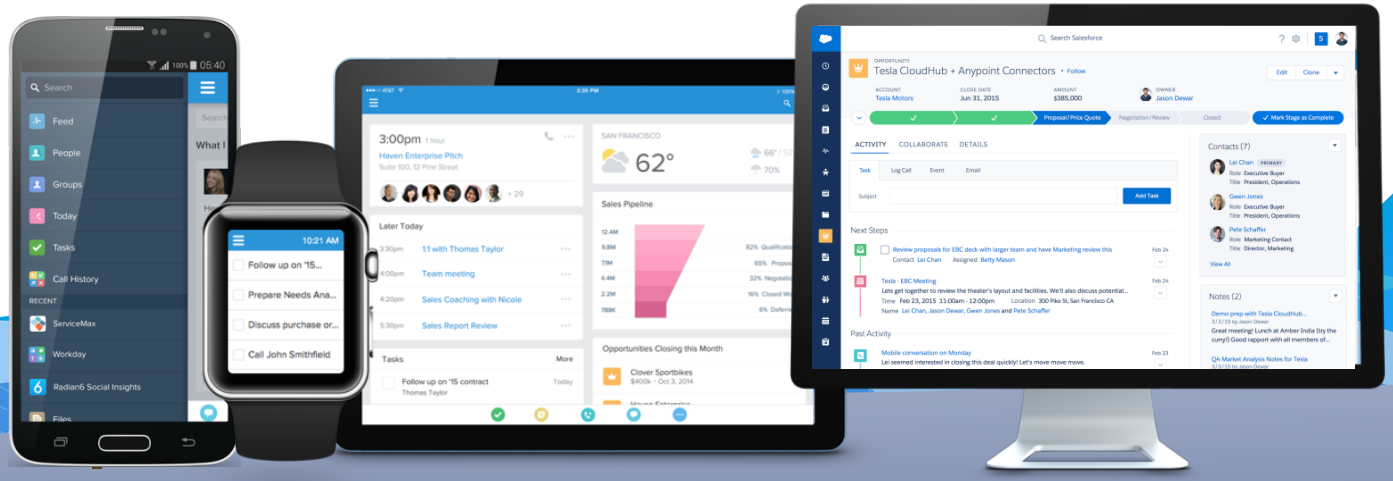
An intuitive, modern experience to drive productivity on every device

Lightning Builder

Build, customize and deploy apps faster than ever, no code required

Lightning Ecosystem

Discover and use components from a vibrant ecosystem



Lightning Experience for End Users

A connected, intuitive user experience on any device

Work Faster

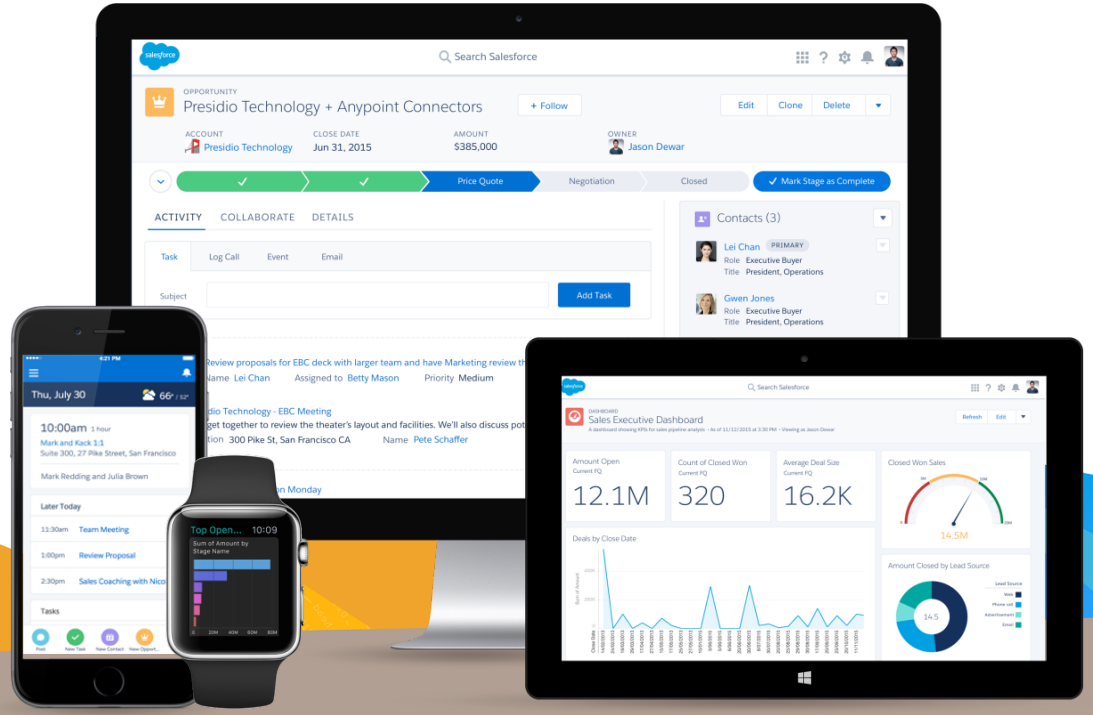
Easy-to-use app experience to maximize productivity

Work Smarter

Analytics and in-line intelligence to drive precision

Work the Way You Want

Get a customized, consistent experience across every device



Empower Everyone to Build Faster with Lightning Builder

Build, customize and deploy apps faster than ever, no code required

Easy to customize

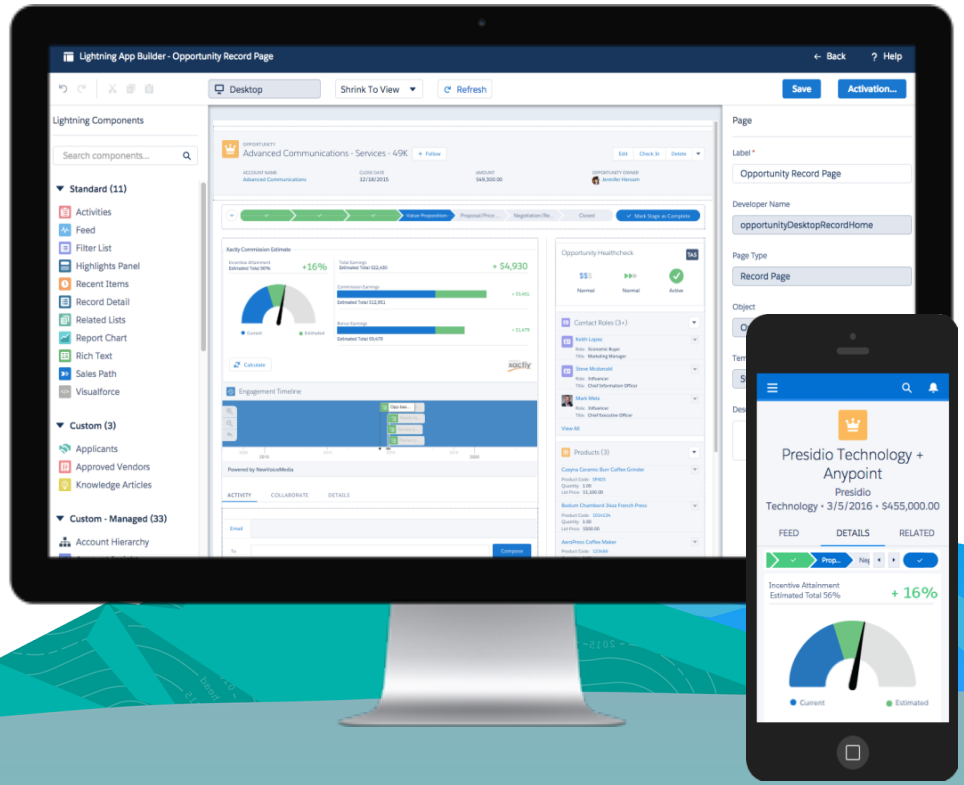
Drag and drop to make changes. Get end user feedback and iterate, instantly.

Build apps faster than ever

Customize pages in the Lightning Experience and build apps without code, for any device.

Everyone is empowered to build apps

Developers code and share components, while admins drag-and-drop components into pages.



Build and Customize Lightning Components

Democratize App Development with UI/UX component framework

Lightning Framework

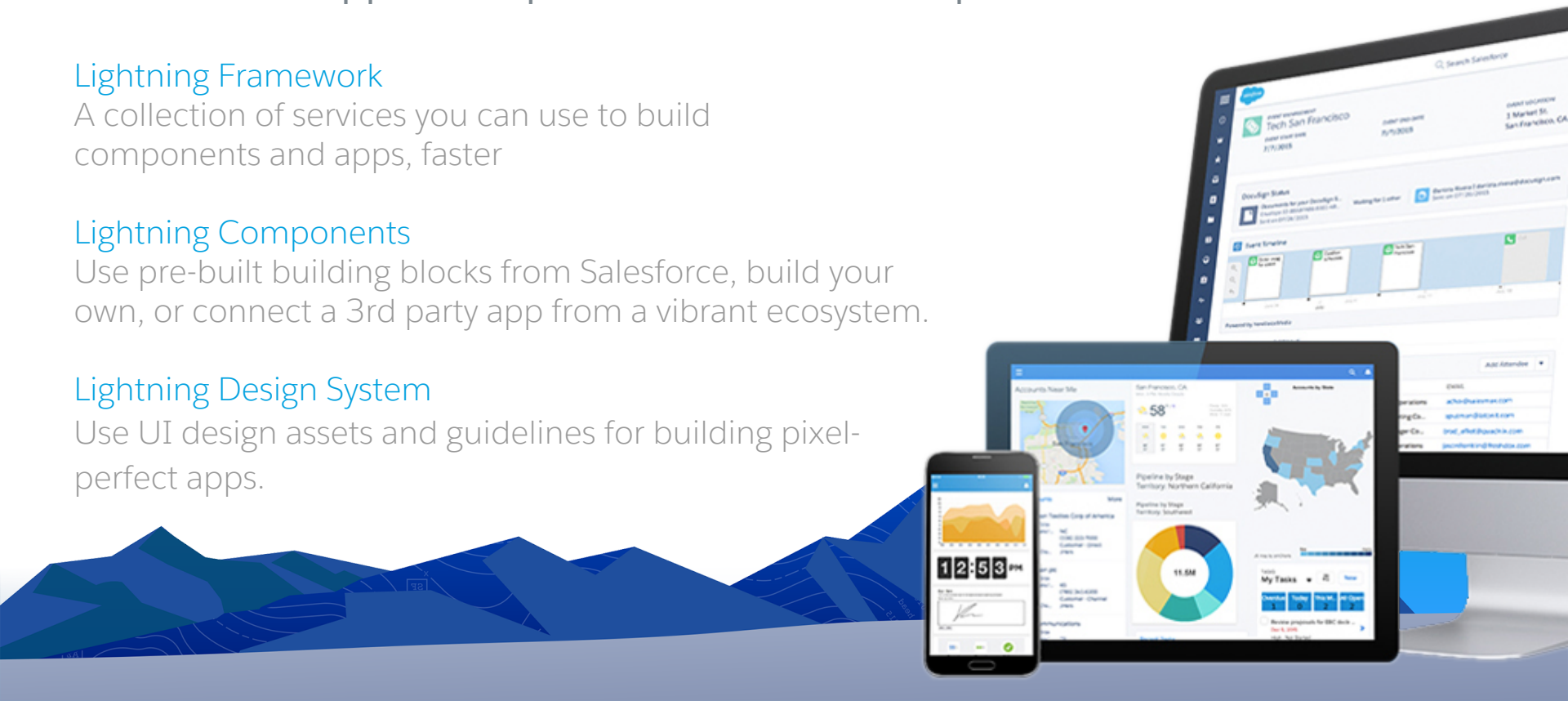
A collection of services you can use to build components and apps, faster

Lightning Components

Use pre-built building blocks from Salesforce, build your own, or connect a 3rd party app from a vibrant ecosystem.

Lightning Design System

Use UI design assets and guidelines for building pixel-perfect apps.



Lightning Delivers a New Continuum of App Development

Ensuring Scalability, Performance, Availability

Developers ⟷ Admins ⟷ End users →

Lightning Components	Lightning Design System	Lightning Exchange	Lightning App Builder	Lightning Community Builder	Lightning Process Builder	Salesforce 1 Mobile App	Lightning Desktop Experience	Customer, Employee, Partner Community
								
Build	Style	Buy	Customize	Reuse	Automate	Use everywhere		

Lightning Components Overview

What are Lightning Components ?

- Components are self contained, re-usable units of an app
- Build complex apps using multiple simple components - including **Base Lightning Components**
- Inter-component interaction happens through events, attributes and methods
- Client side UI rendering rather than server side

What is Lightning Component Framework?

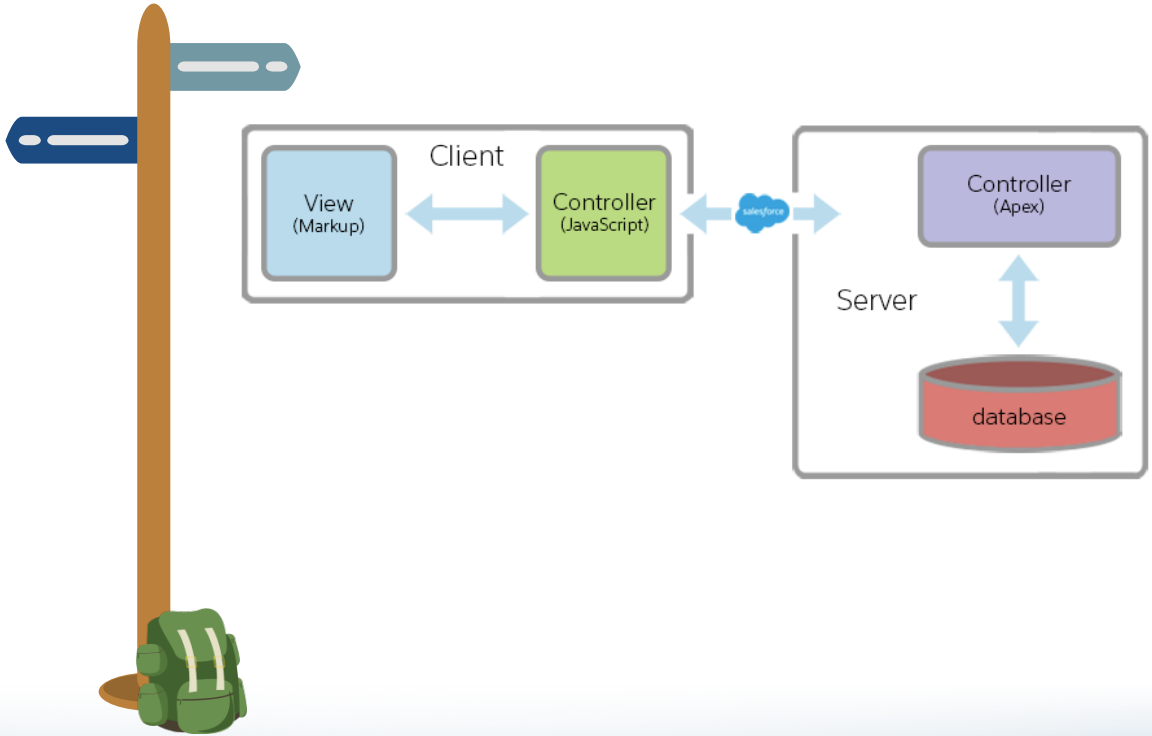
A modern UI Framework for developing dynamic single page web apps engineered for growth

Built on the Open Source Aura Framework

Framework for both mobile and desktop devices

Multi-tier component development bridges client and server

Ecosystem supported by AppExchange



Why Lightning Component Framework?

Features of Lightning that provides competitive advantage

Out-of-the-box set of components, NATIVE to Salesforce

Optimized for performance

Loosely Coupled with publish/subscribe event driven architecture

Extendable and Reusable across apps and orgs

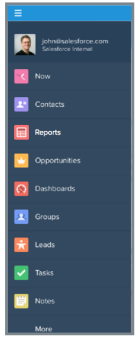
Portable via Lightning Out : Build once and use anywhere

Responsive UI

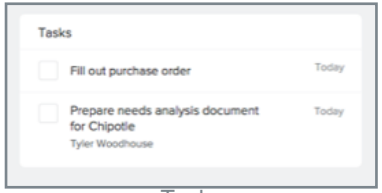
Built on metadata from the foundation, accelerating development

Lightning Components come from 3 places

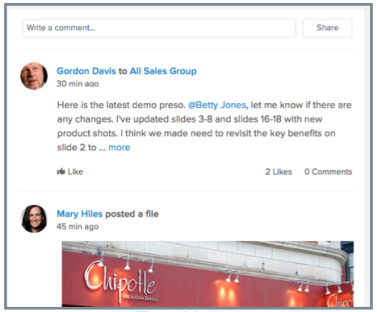
Built by Salesforce



Left nav



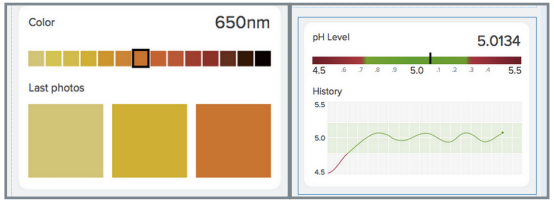
Tasks



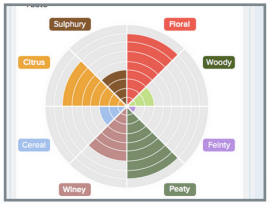
Feed items

Publisher bar

Built by partners, developers, and customers

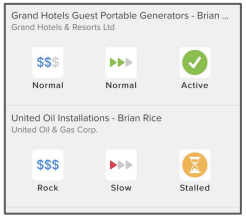


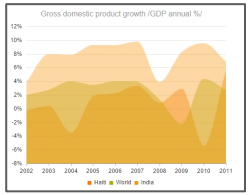
Sliders

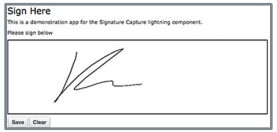


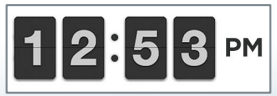
Multi-view charts

From the AppExchange



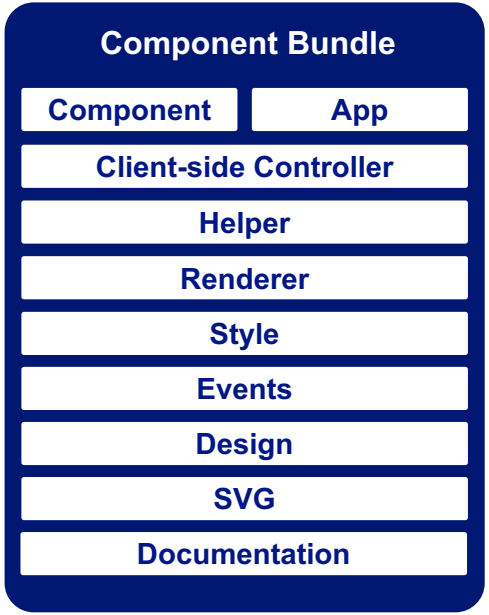






Anatomy of a Lightning Component

Self contained, reusable UI bundles

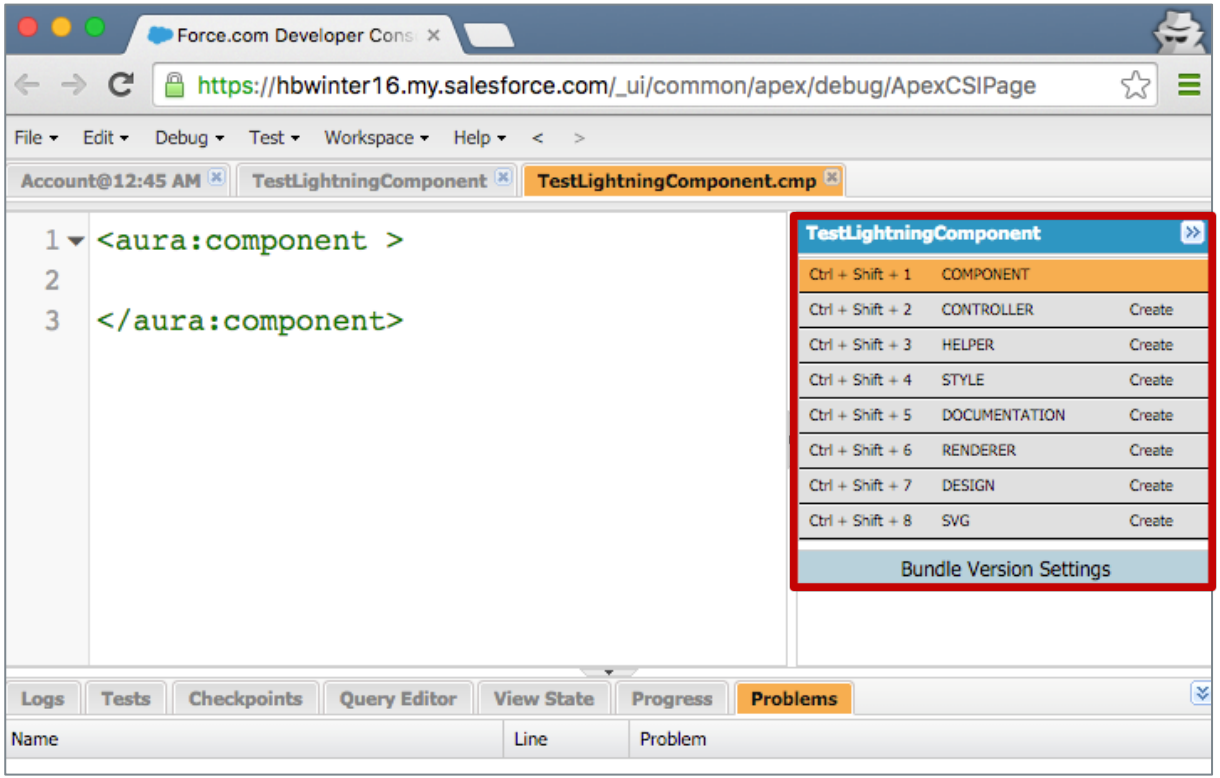


Client-side
(runs in the client device web browser/web-view)

Server-side
(runs in the salesforce platform)

- Bundle is the development/deployment container
- Component & App contains the markup; one per bundle – either a component or app, not both
- Client-side controller contains the application logic in form of actions, coded in JavaScript; unique per instance of component
- Helper contains reusable JavaScript code that is shared across multiples instance of the same component at runtime
- Renderer contains any custom rendering (DOM manipulation) code for the component; not required in most cases
- Style contains any custom CSS code for the component
- Events (Component/Application) enable intra/inter component communication using publish/subscribe model
- Design contains labels and additional descriptions for attributes exposed in Lightning App Builder
- SVG contains the vector graphics for component icon that is visible in Lightning App builder design panel
- Documentation contains the component documentation
- Apex controller for server side logic; maximum of one per component

Developer Console



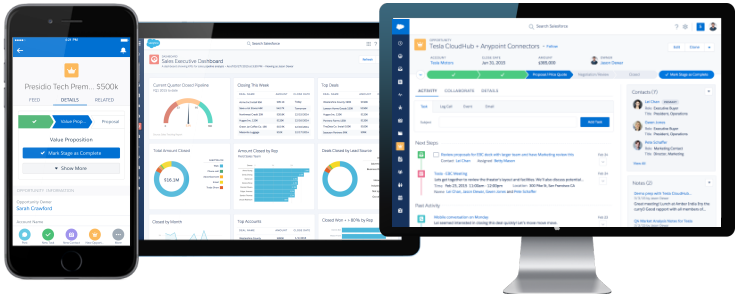
Lightning and Visualforce are fundamentally different

LIGHTNING	visualforce
CLIENT-SIDE UI GENERATION	SERVER-SIDE UI GENERATION
APP-CENTRIC MODEL	PAGE-CENTRIC MODEL
COMPONENT BASED FRAMEWORK	MVC FRAMEWORK
JAVASCRIPT FRAMEWORK	TAG-BASED LANGUAGE
DESIGNED FOR LIGHTNING EXPERIENCE AND SALESFORCE1	DESIGNED FOR SALESFORCE CLASSIC

Credit: Mike Topolovich, Patterns for Migrating from Visualforce to Lightning Components: Part 2

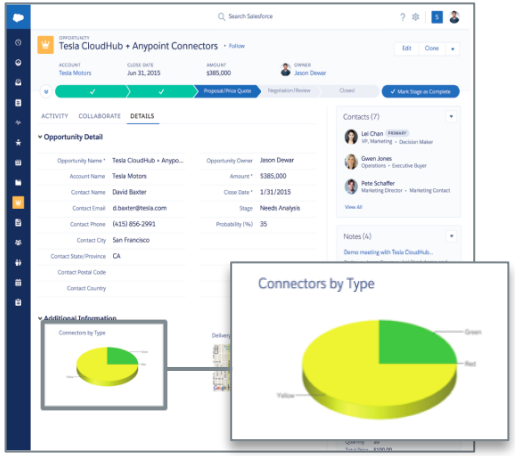
Visualforce will be supported for the long term

Lightning is the future of Salesforce technology



Both our mobile and new desktop are built with Lightning Components

Visualforce and your customizations will continue to be supported in Lightning Experience*

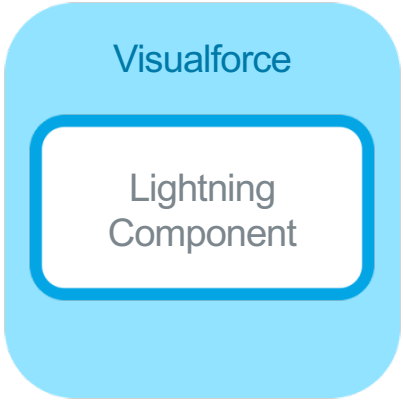


*see exceptions list

Visualforce and Lightning work together

Visualforce will always be supported. Start planning for Lightning.

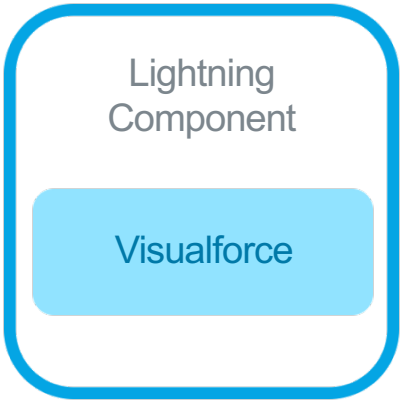
1.
Use Lightning Components in Visualforce



"Lightning for Visualforce"

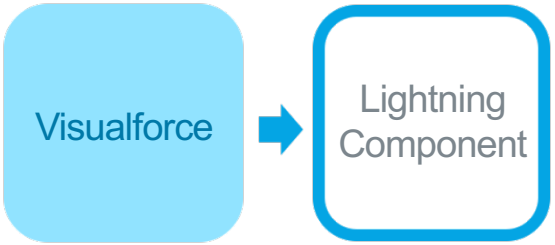
Winter '16

2.
Use Visualforce in a Lightning Component



Visualforce as an iframe

3.
Migrate Visualforce to Components



Lightning Components will have the same functionality as Visualforce

Coming soon

https://www.youtube.com/watch?v=6uQPV7Xf9PI

Lightning Components Usage

Lightning Pages

- Add components to Lightning App pages.
- Includes record detail pages and a home page.
- Email Application Panes (beta)

Salesforce1

Add components to Lightning Component tab.

Standalone Apps

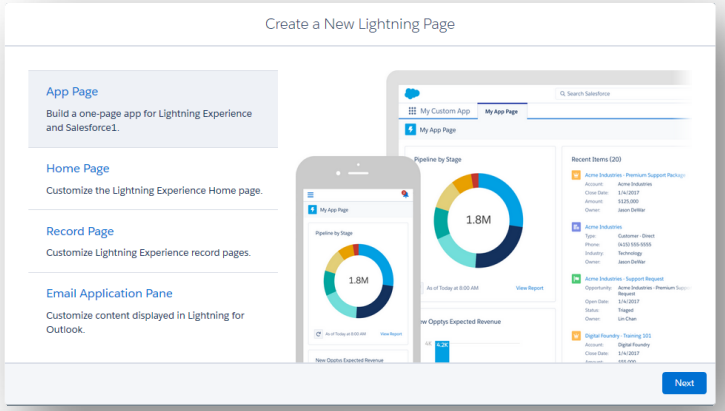
Add components to full-screen apps.

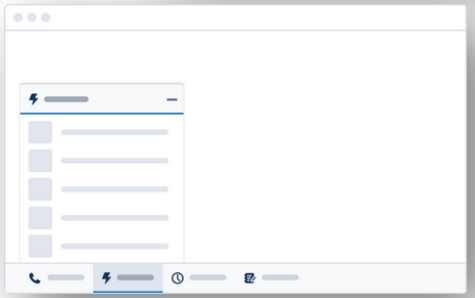
Lightning App Utility Bar

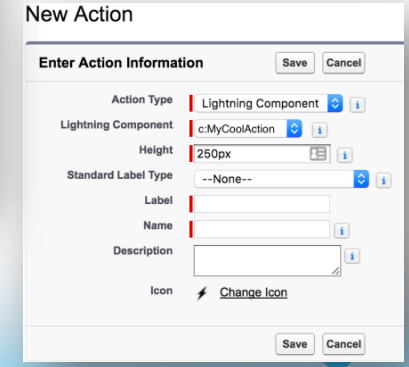
Add components to the Utility bar of Lightning Apps.

Lightning Actions

Add components as actions to Lightning pages.





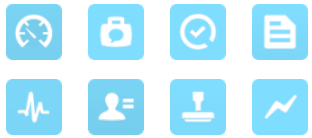


Empower Everyone with the Lightning App Builder

Now Anyone Can Build Apps for Every Device

Drag & Drop

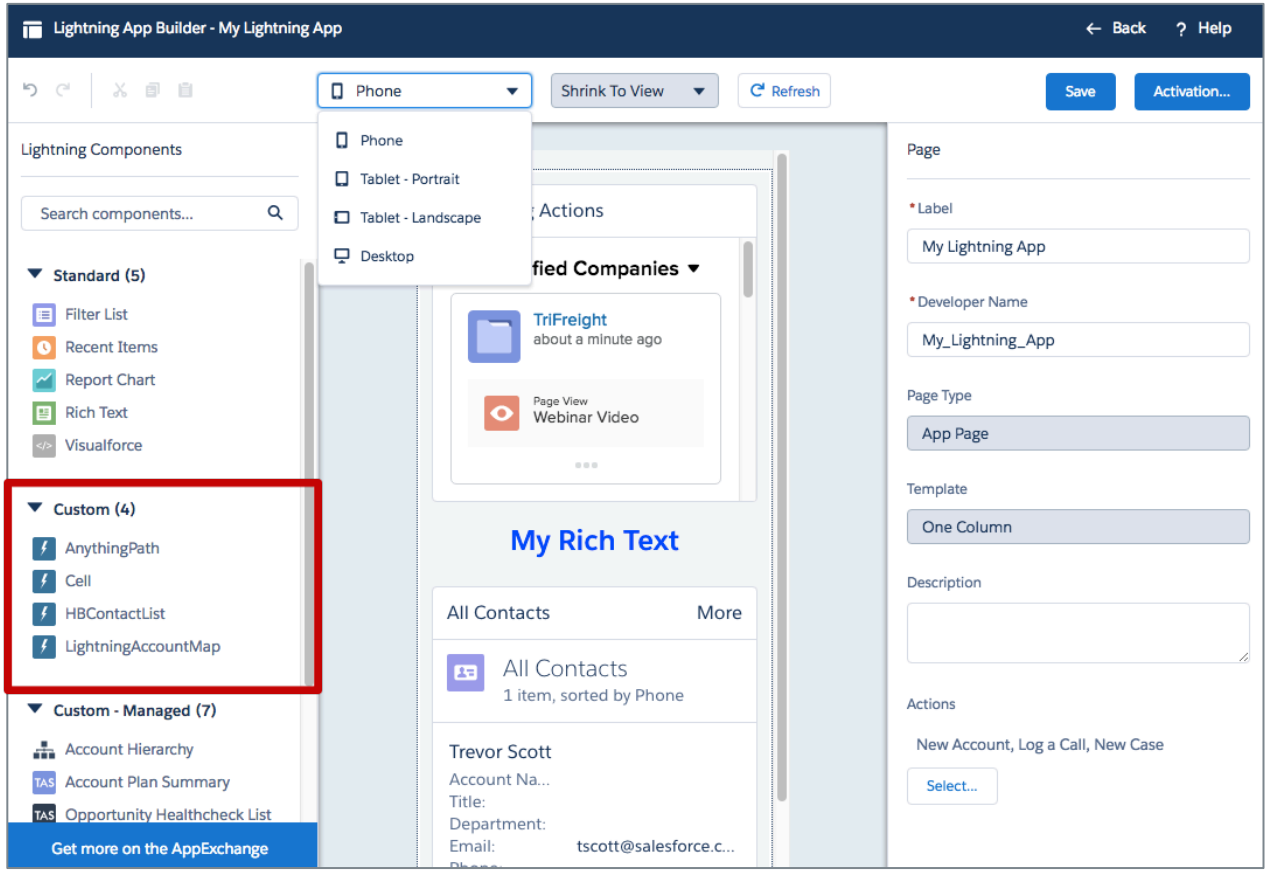
Build with standard and custom Lightning Components



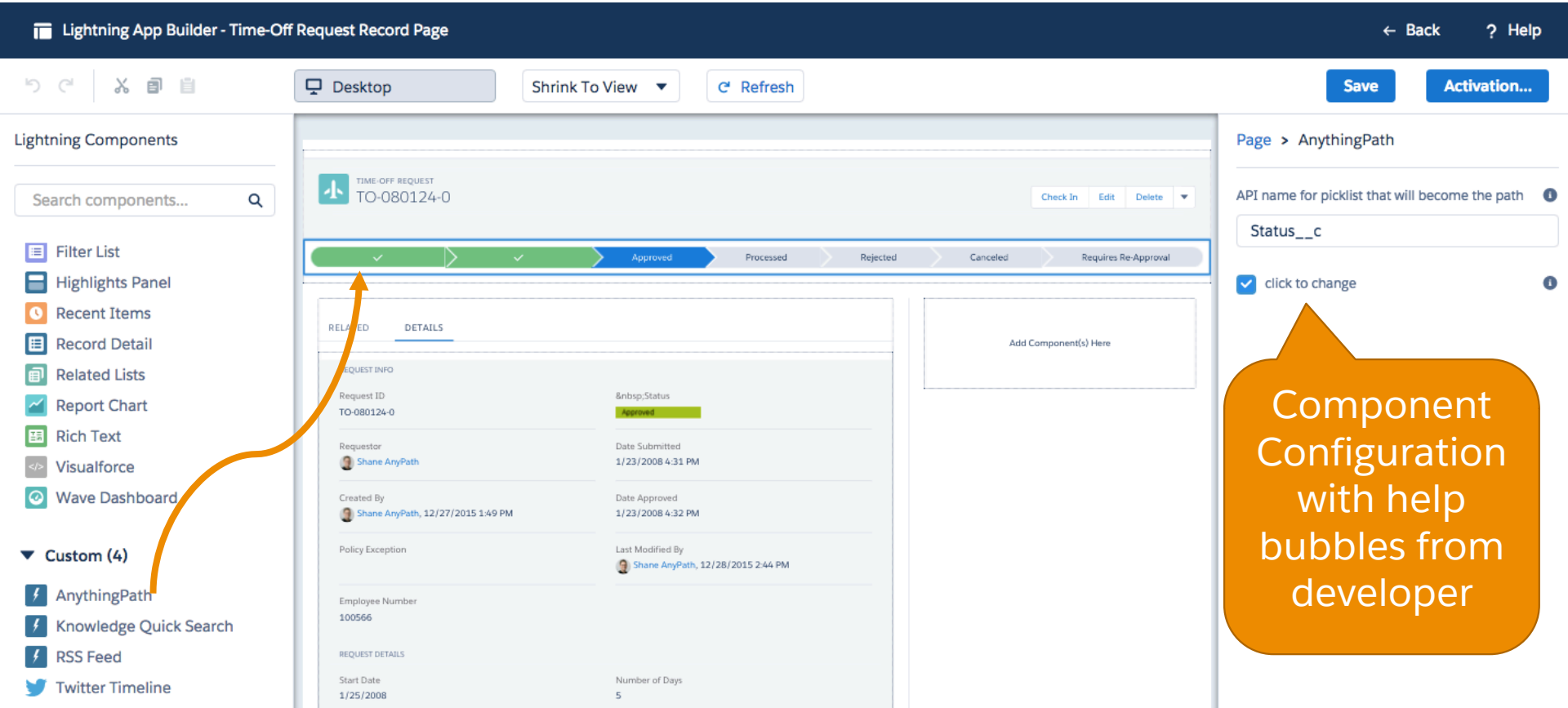
Any Device

Design apps for every screen from one canvas

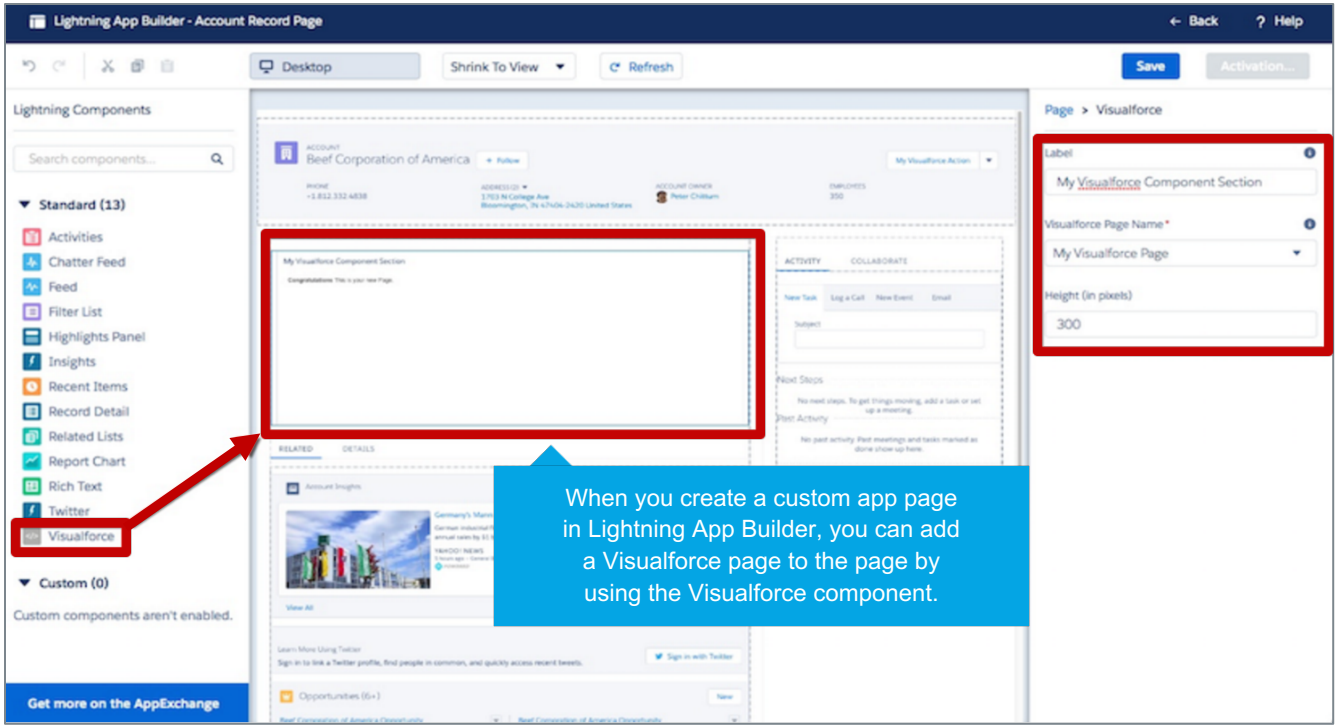
Lightning Components



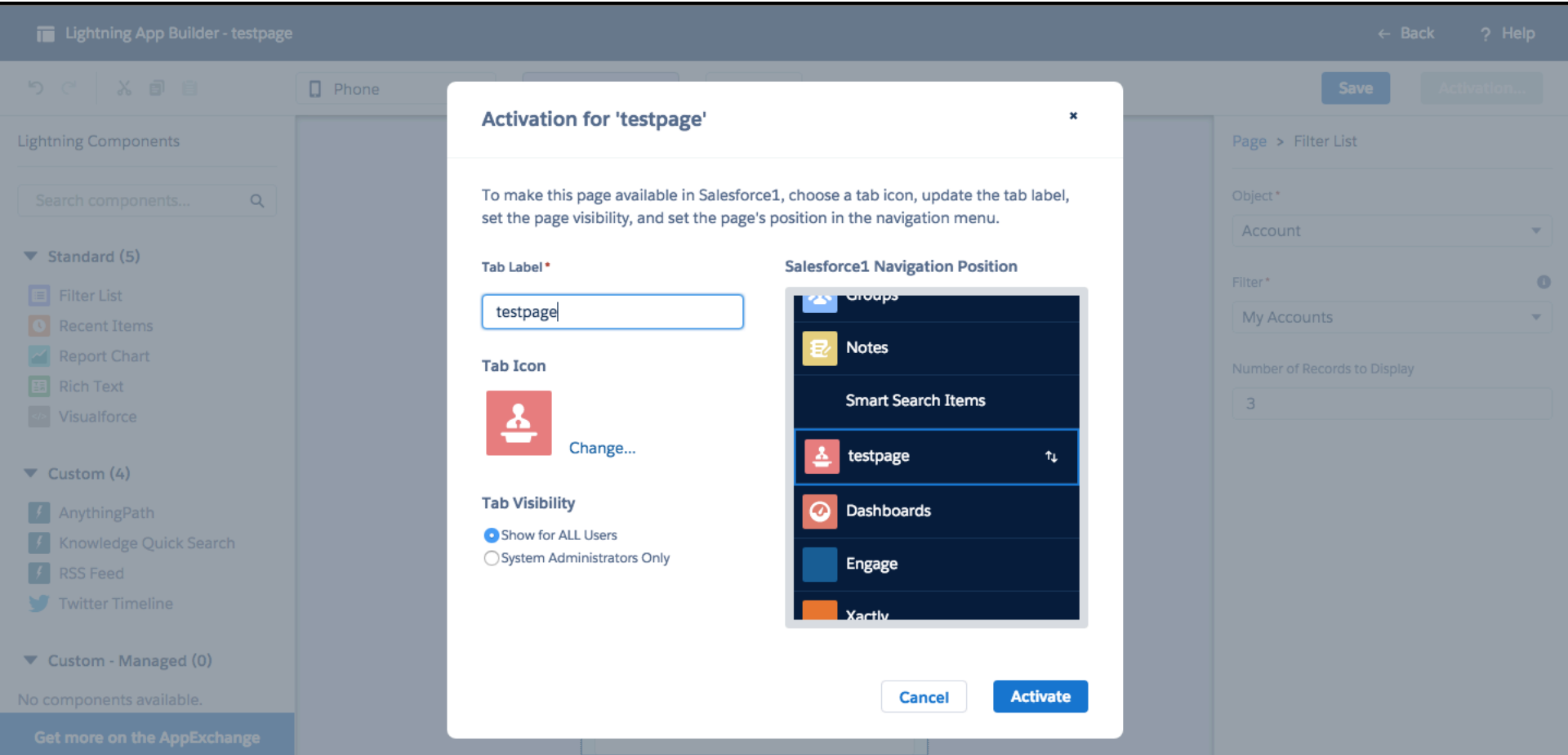
Key Capabilities: LEX Record Page



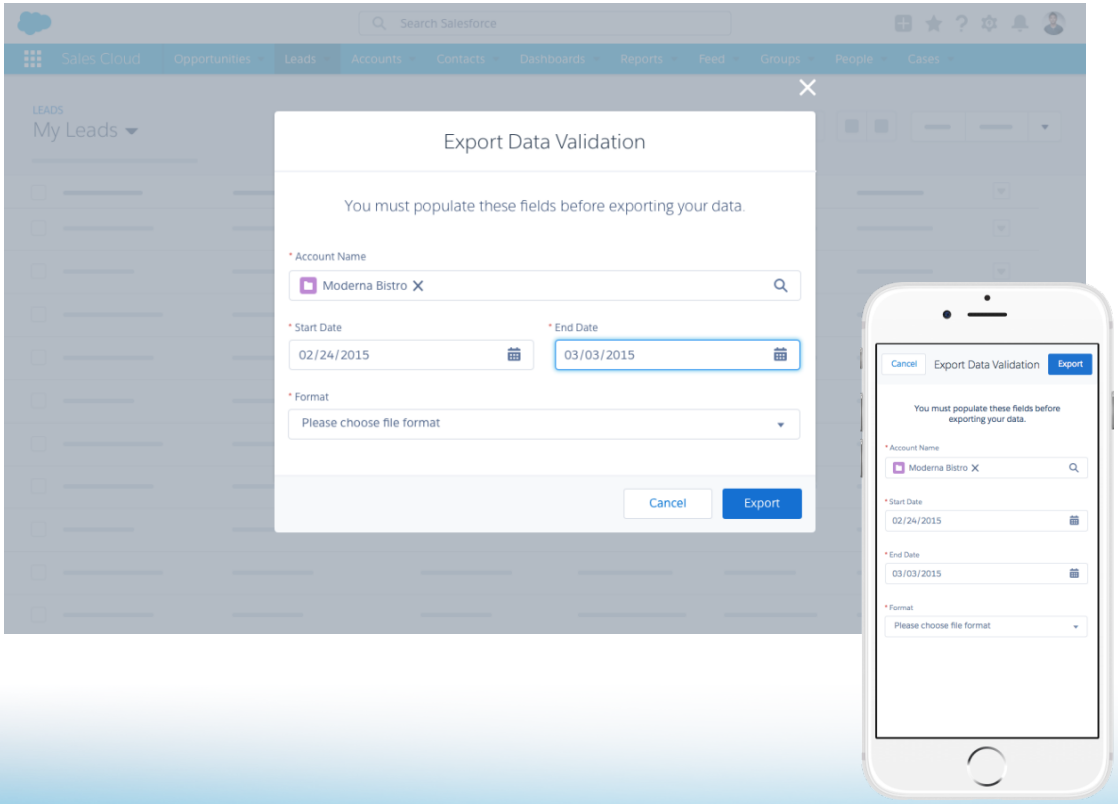
Lightning Supports Visualforce



Key Capabilities: Salesforce1 Tabs



Lightning Actions



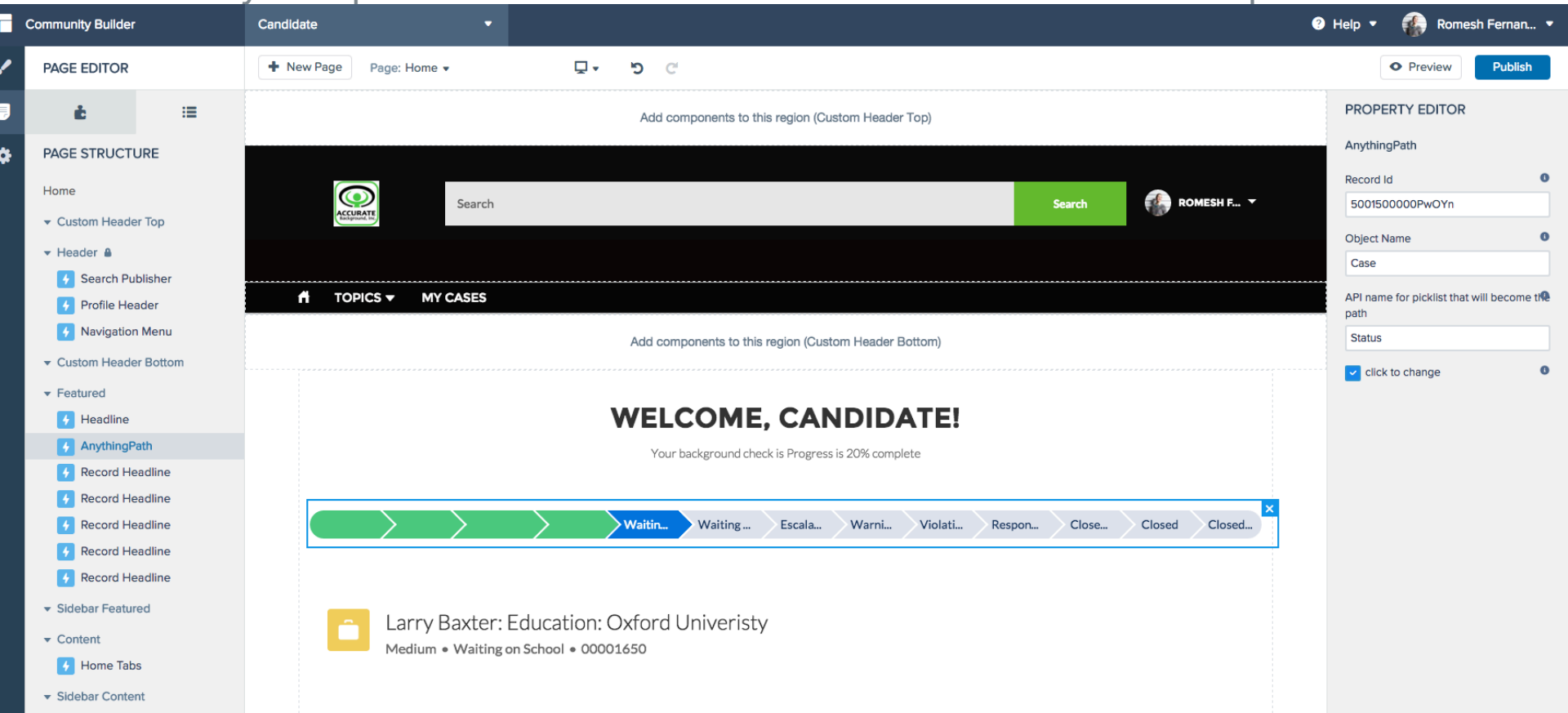
Invoke a custom Lightning Component in Salesforce1 and Lightning Desktop

Custom Global Actions (Spring ‘17)

Custom button, custom link, mass action support (Future)

Key Capabilities: Community Builder Extensibility

Community Templates use a similar UI and can use the same components



Extras

What is the Lightning Data Service? (GA Summer '17)

A centralized data and caching service for Lightning

Benefits:

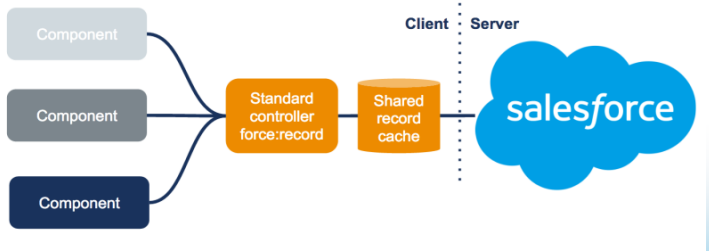
Developer delight
No APEX. No SOQL. Auto handling of FLS and sharing. Declarative CRUD operations.

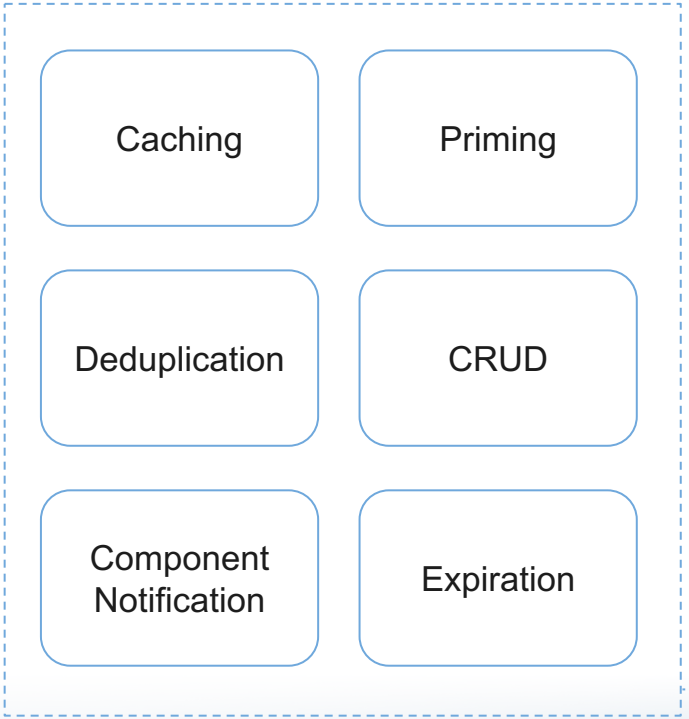
Performance
Cache hits and shared data requests improve performance.

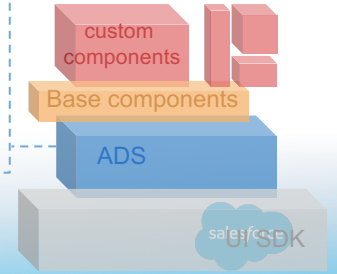
Single data source
Different components using the same records always display the same data.

Notifications when data changes
recordUpdated event fired on force:record when data changes.

Offline read and write
Local drafts of saves and deletes persist on device until connection is restored.







Locker Service In Lightning Components

- LockerService is a powerful **security architecture** for Lightning components. LockerService enhances security by **isolating individual Lightning components in their own containers**. LockerService also promotes best practices that improve the supportability of your code by only allowing access to supported APIs and eliminating access to non-published framework internals.
- LockerService Requirements:
  - JavaScript ES5 Strict Mode Enforcement
  - DOM Access Containment
  - Restrictions to Global References
  - Access to Supported JavaScript API Framework Methods Only
  - Stricter Content Security Policy (CSP)

Lightning LockerService

Rollout:

Winter ‘17: Critical Update for ALL Orgs

Spring ‘17: Critical Update for All Orgs + CSP Checks in Sandbox and DE Orgs

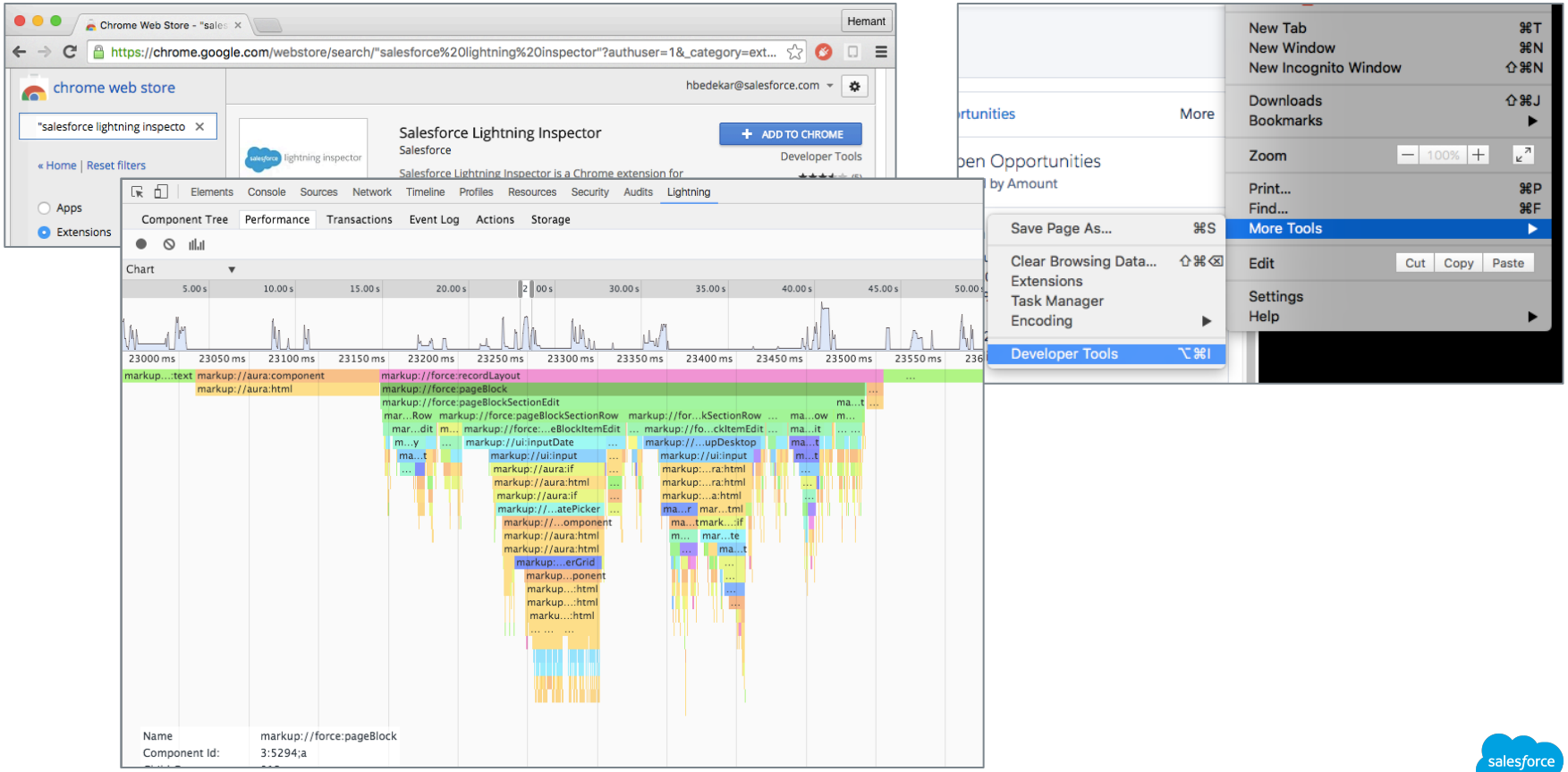
Summer ‘17: Auto-Enabled in ALL Orgs

* For components of API version 40 and above

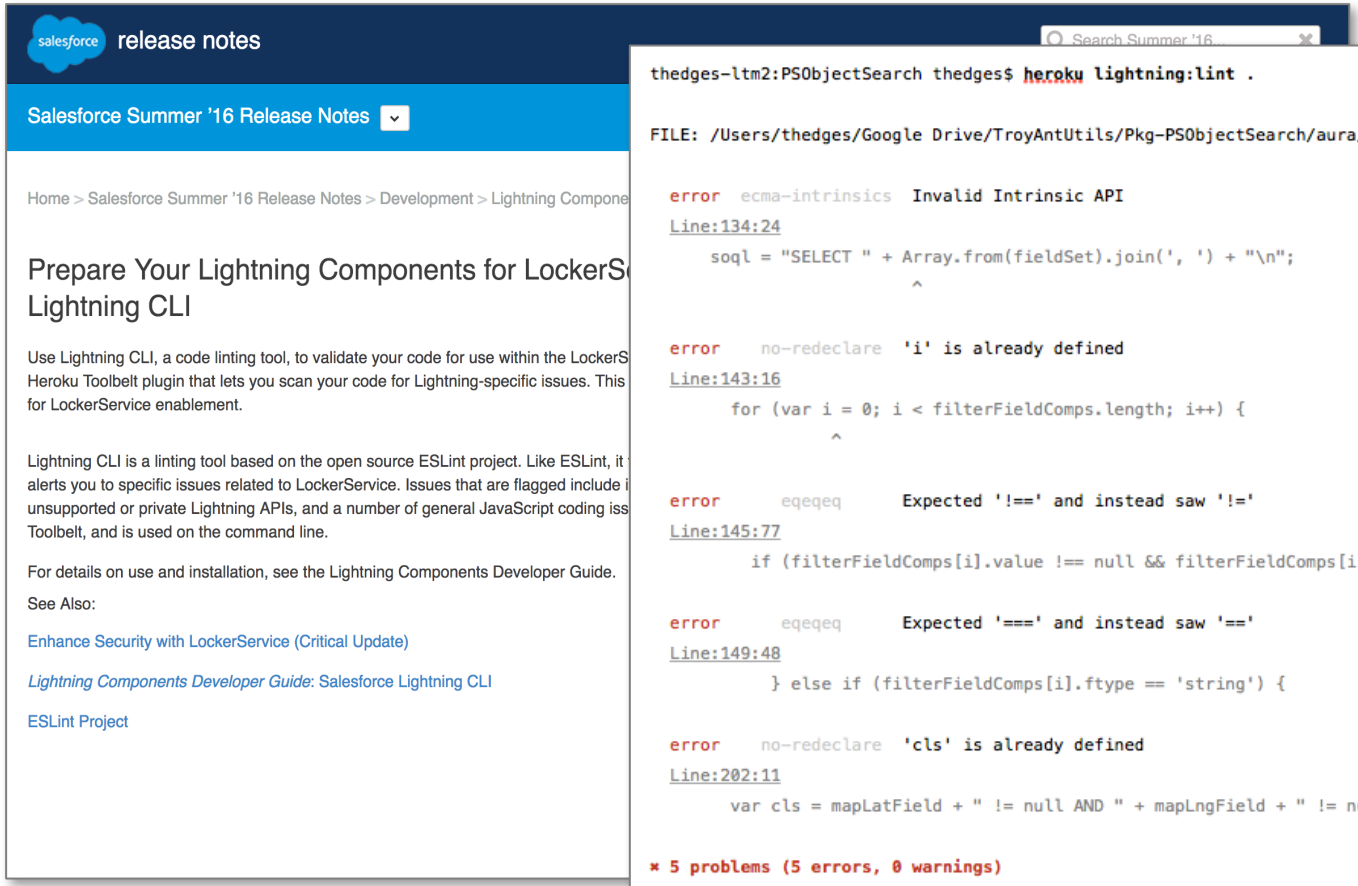
How to Debug Lightning Components

- Use JS debugger, console.log statements
- Use Lightning inspector plugin in chrome
- Use Lightning CLI to debug code (LINT)

Lightning Inspector



Heroku Toolbelt: Lightning Lint




```
thedges-ltm2:PSObjectSearch thedges$ heroku lightning:lint .

FILE: /Users/thedges/Google Drive/TroyAntUtils/Pkg-PSObjectSearch/aura/PSObjectSearch/PSObjectSearchHelper.js:

  error  ecma-intrinsics  Invalid Intrinsic API
  Line:134:24
      soql = "SELECT " + Array.from(fieldSet).join(', ') + "\n";
                         ^

  error    no-redeclare  'i' is already defined
  Line:143:16
        for (var i = 0; i < filterFieldComps.length; i++) {
                 ^

  error      eqeqeq      Expected '!==' and instead saw '!='
  Line:145:77
          if (filterFieldComps[i].value !== null && filterFieldComps[i].value != "") {
                                                                              ^

  error      eqeqeq      Expected '===' and instead saw '=='
  Line:149:48
           } else if (filterFieldComps[i].ftype == 'string') {

  error    no-redeclare  'cls' is already defined
  Line:202:11
        var cls = mapLatField + " != null AND " + mapLngField + " != null";

✖ 5 problems (5 errors, 0 warnings)
```

Resources

Create Modern App UI/UX with Lightning Design System

Developers can focus automating business processes rather than creating pixels

Consistent User Experience

Salesforce style guides and modern enterprise design patterns give a unified experience across every app

Deliver Apps Fast

Open source framework and reusable code base to build apps in record time

Seamless upgrades across releases

Easy migration path with Design Tokens versioned for each release

https://www.lightningdesignsystem.com/

Trailhead: Lightning

7 hrs 15 mins
7 Modules

Get Started with Lightning Experience

Start your journey to becoming a new Salesforce Admin with Lightning Experience.

Beginner | Administrator

3 hrs 45 mins
4 Modules

Migrate to Lightning Experience

Increase the productivity of your sales team by migrating your existing org to the new Lightning Experience.

Intermediate | Administrator

1 hr 55 mins
2 Modules

Sell Lightning Fast with Salesforce

Learn how to use the new Salesforce to sell lightning fast in as few clicks as possible.

Beginner | Business User

10 hrs 35 mins
5 Modules

Develop for Lightning Experience

Build apps fast for Lightning Experience with Visualforce, components, and new design resources.

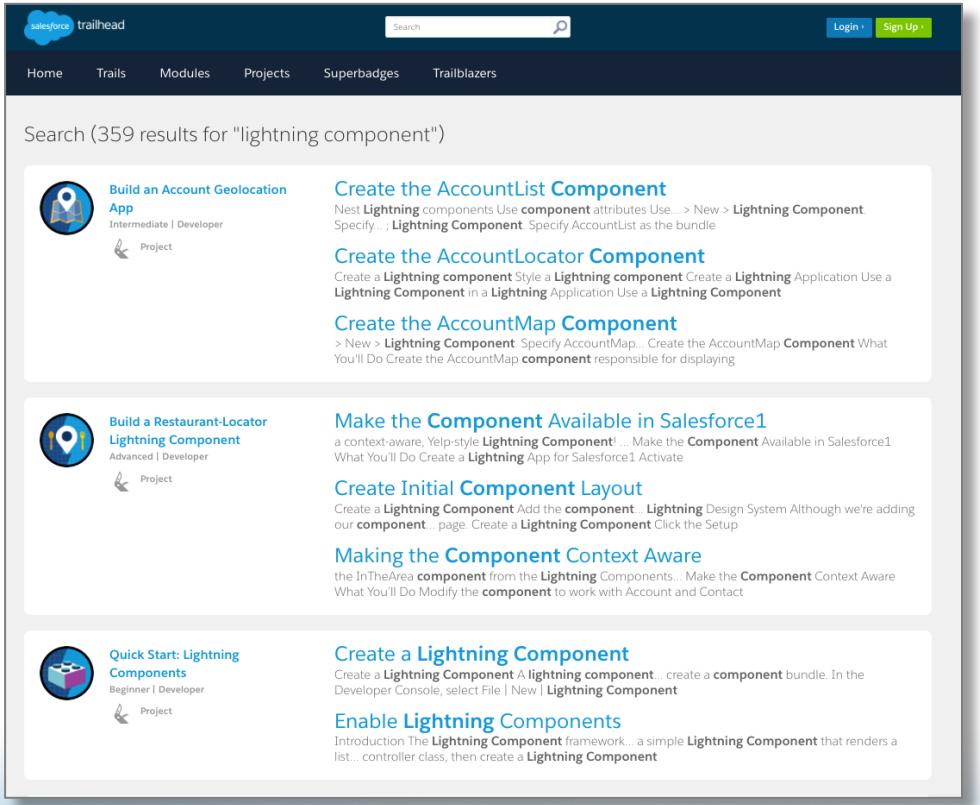
Intermediate | Developer

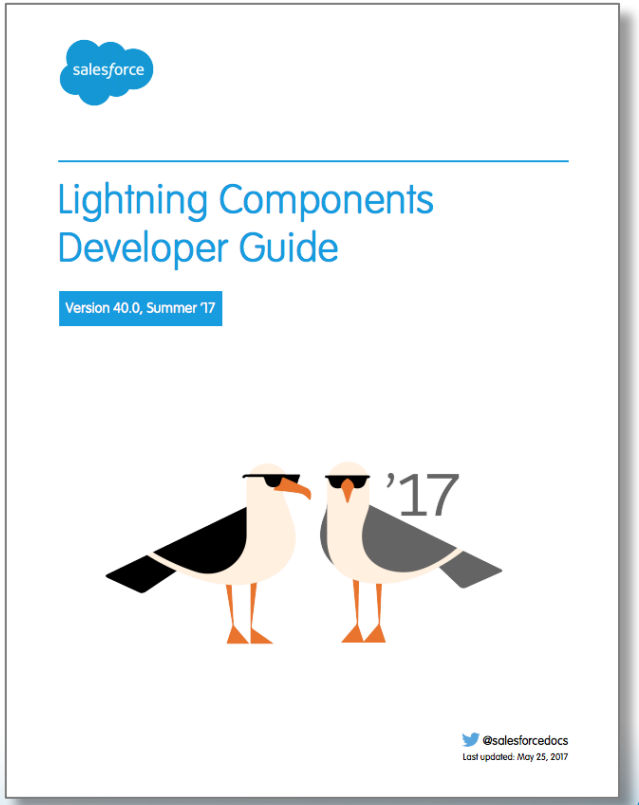
Unlocked

Lightning Experience Specialist

Power up your sales reps with a super-charged interface and process automation.

Learn How to Build Lightning Components

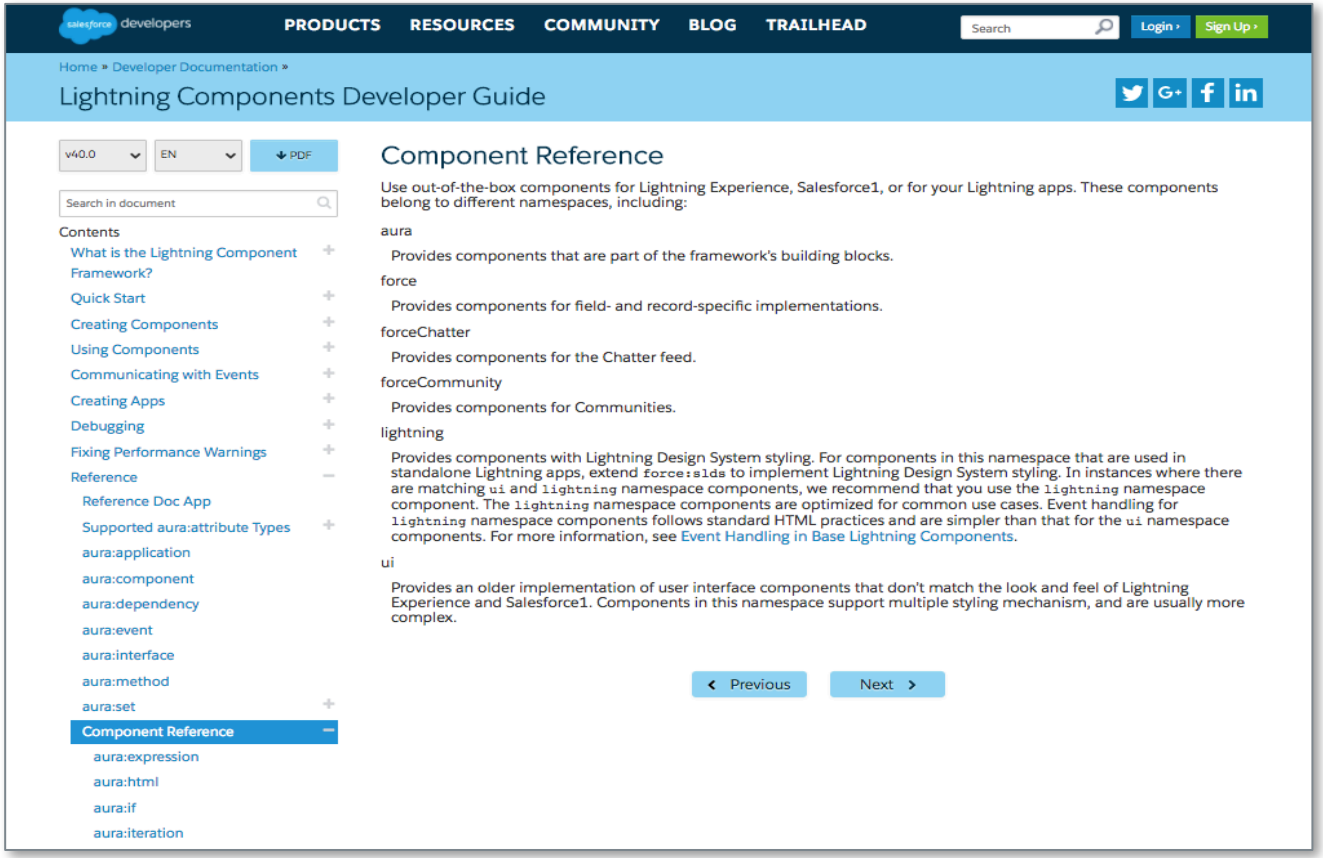




https://trailhead.salesforce.com/

salesforce

Lightning Component Development Guide



https://developer.salesforce.com/docs/atlas.en-us.lightning.meta/lightning/aura_compref.htm

Introducing AppExchange for Components

The Enterprise Marketplace for Lightning Components

Build apps easily

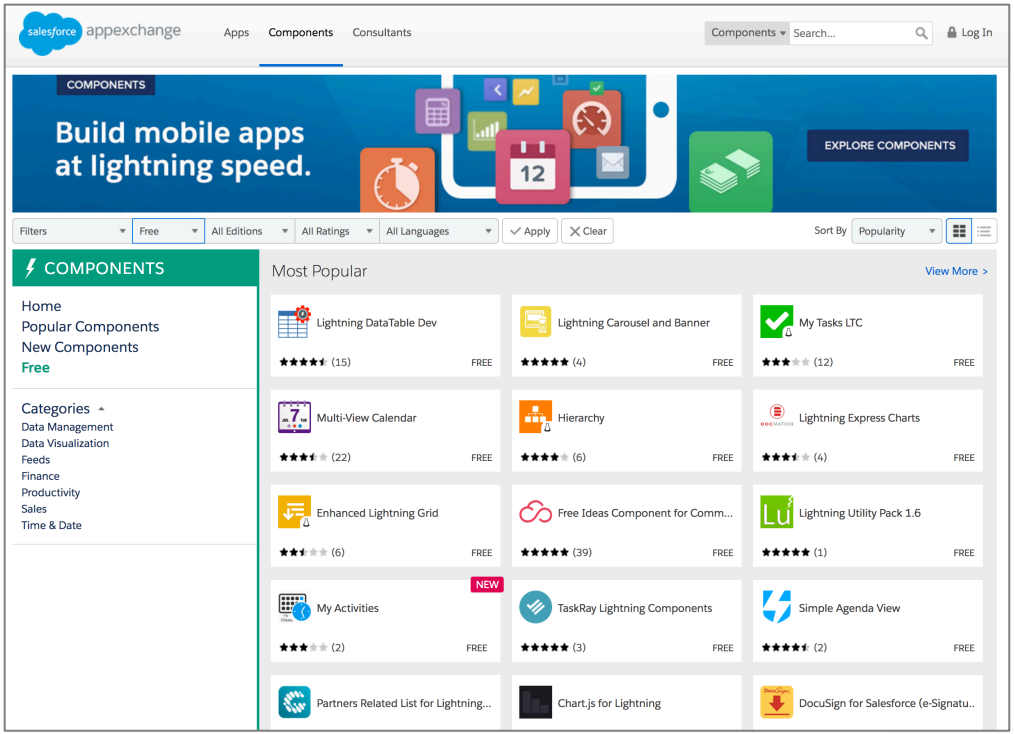
With pre-built Components from Salesforce partners that work for any device

Build More Engaging Apps

Use the best pre-built components in your apps

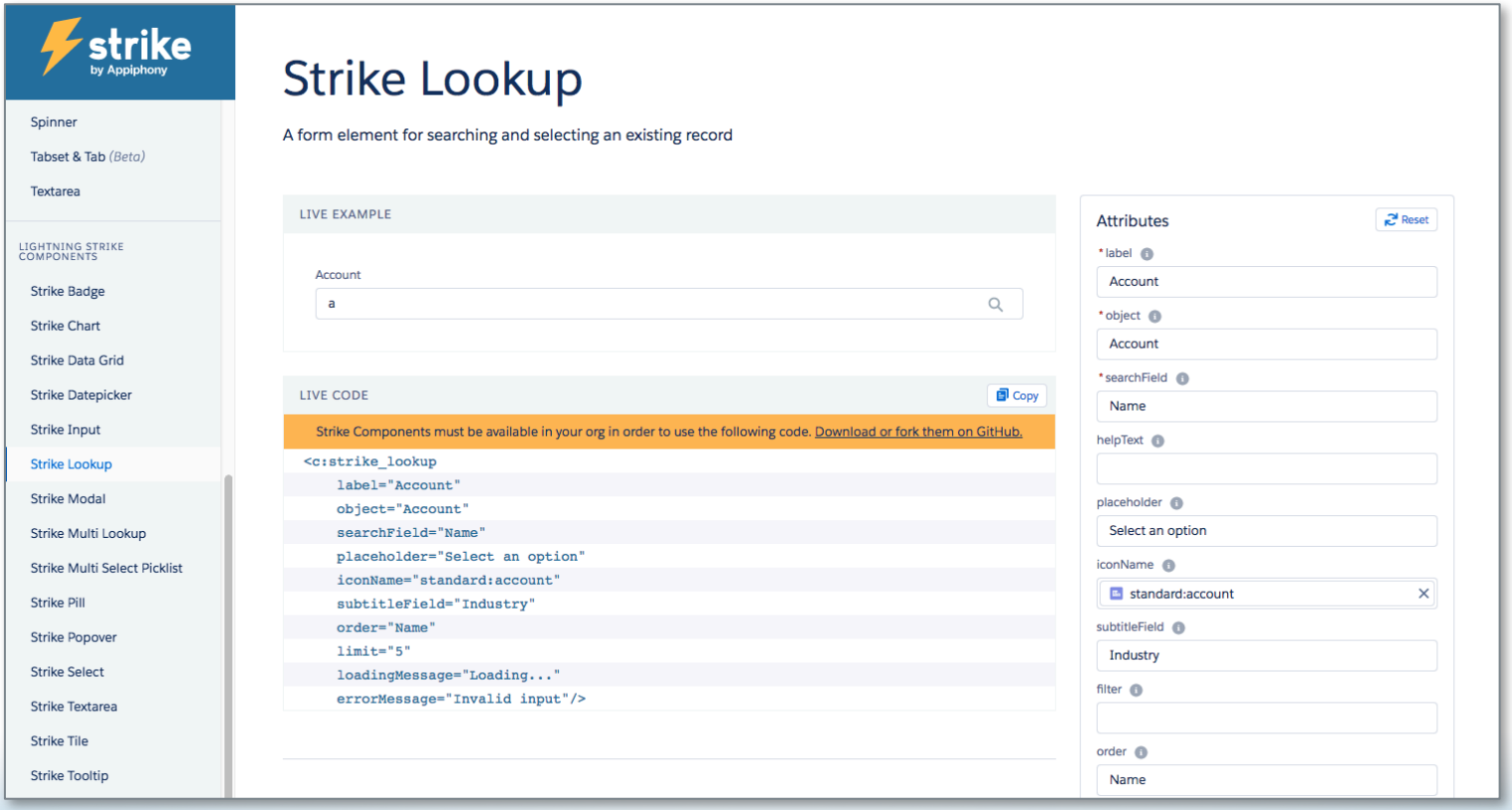
Deliver apps faster

Use Components from AppExchange with your own custom-built Components



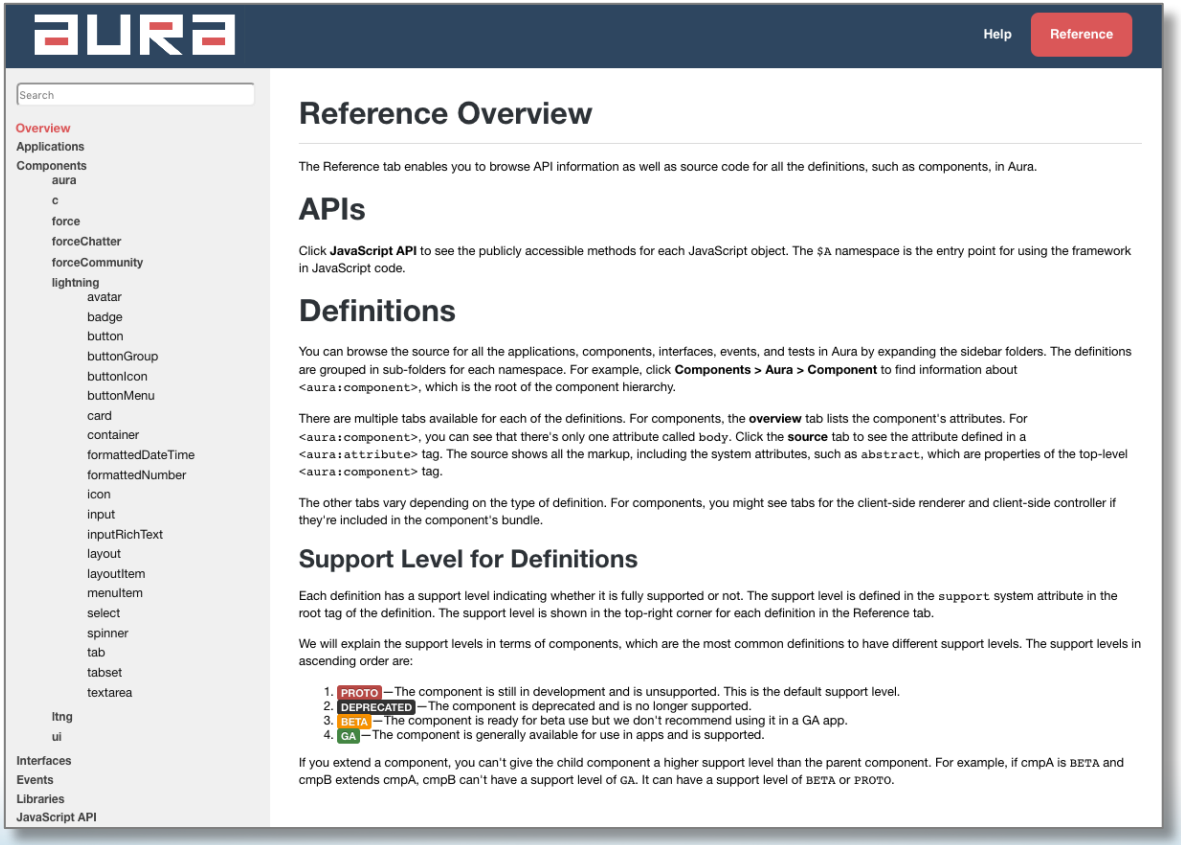
https://appexchange.salesforce.com/components/free

Strike Components by Appiphony



http://www.lightningstrike.io/

Lightning Component Reference



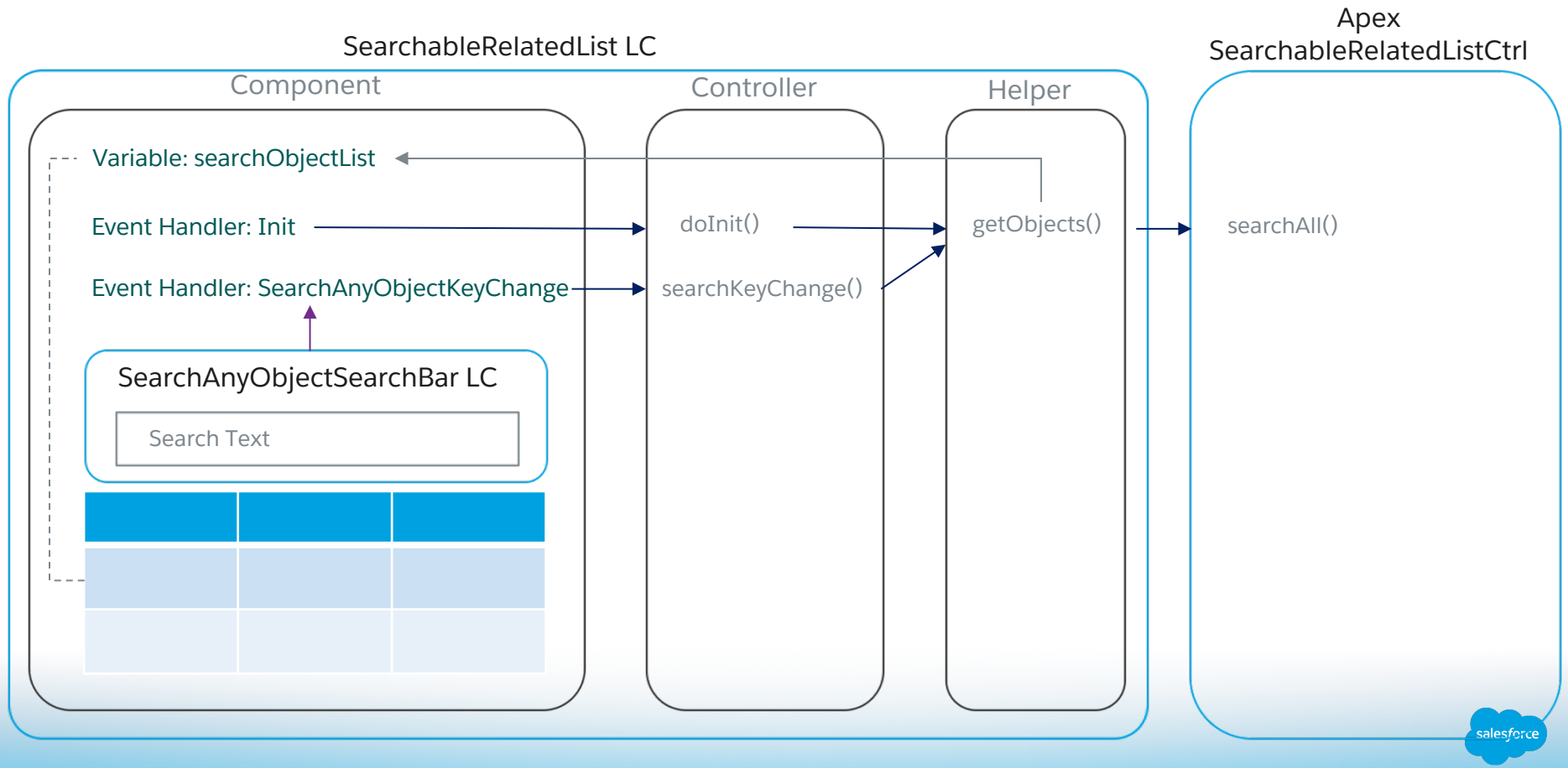
https://<org-url>**/auradocs#reference**

GitHub

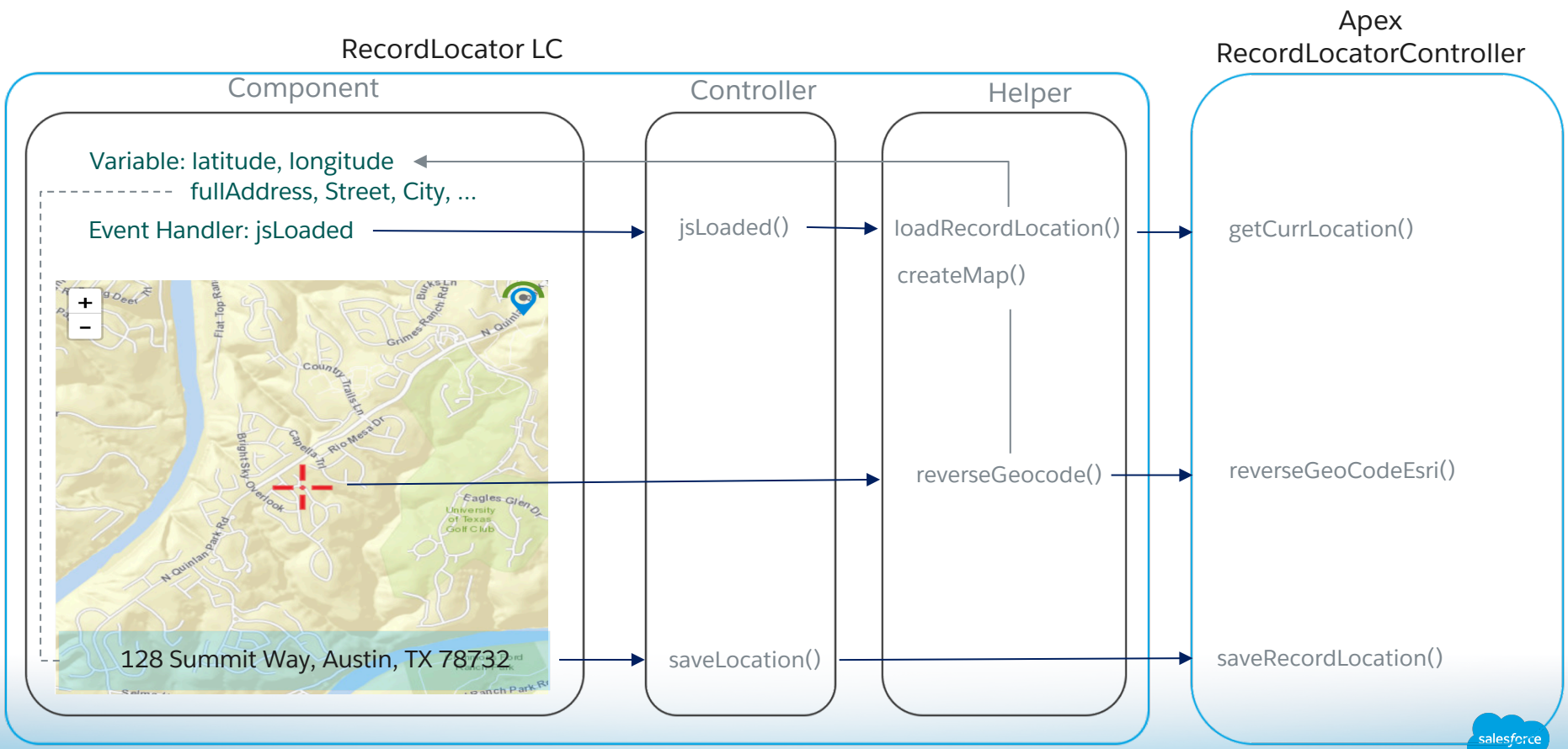
- Christophe Coenraets - https://github.com/ccoenraets
  - DreamHouse Sample App - https://developer.salesforce.com/dreamhouse/
- Shane McLaughlin - https://github.com/mshanemc/
- Troy Hedges - https://github.com/thedges

Demo

Searchable Related List



Record Locator



Thank you