

Summer '20 Abridged

The [Salesforce Discord Collective](#) Presents:

THE SUMMER 20 RELEASE NOTES - ABRIDGED

Best read from 1.5m distance

CRITICAL STUFF

- [Lightning Sync](#) will be [retired in Winter 21](#). Recommended integration path: Einstein Activity Capture. This is the what, [5th release in a row](#) they've [nuked some integration](#) with Google/Microsoft ? At this point I'd probably just recommend going third-party.
- [Pardot Connector](#) needs to be manually updated to v2.
- [After Save Flows](#), [Platform Event Flows](#), and [Flows that Bypass User security](#) change [everything](#) about Declarative Automation in Salesforce (but curb your enthusiasm a bit). Read our "Special Edition" at the end for more details, which also covers [Dynamic Forms](#).
- Lightning Message Service (`LMS`) is [now GA-ed and available](#). This **replaces** the janky `pubsub` module found with `lwc-recipes` and is now the official communication mechanism between VF pages, Aura, LWCs, and components placed in the lightning utility bar. See [/u/temp_sv_dev](#) 's [github repo here](#) for some samples on how to swap out `pubsub`.
- [Some Permissions Changed](#) and [Need Review](#)

GENERAL STUFF

- [Paths now create Kanban views](#) and are also subject to [confetti](#) which I assume means they've [merged](#) Kanban and Path in the backend
- [Email Insights](#) just got a little bit smarter. Requires Einstein Activity Capture.
- [B2B Commerce Orders protested against confinement](#) and aren't forcibly tied to Products anymore. ~~It's worth a shout because a lot of companies did custom Order__c due to this limitation.~~ Only applies to B2B Commerce, my bad.
- The [Related Record](#) component in the App Builder now allows showing the parent of a parent AND polymorphic relationships, which is actually super nice
- Knowledge [took the red pill](#) and now sees record access as it really is (Beta)
- [Dynamic Actions](#) don't force you to open the Page Layout to changes Actions in Lightning (for desktop)

- [Salesforce Help Login](#) now goes through [trailblazer420.yolo](#) which is OK for most people but annoying for implementation partners which have tons of the same user with the same email.
- [Mobile App](#) team get the “Best Release Notes” award because I can’t summarize them more than they did. Seriously, great job. Also [Custom Notifications](#) get a nice upgrade.

FLOWS

- The Critical stuff we wrote above
- Flow now [try to help you do Flows that suck less](#) with in-app [prompts](#) when you do [things that suck](#)
- Debug Mode is now really a [Debug Mode](#) meaning data isn’t saved when you run the debug thank god finally pfiew
- [Passing Data to Flows](#) now easier
- Developers rejoice as [Apex-defined Variables](#) can now be returned, and [Flows now accurately indicate usage in debug logs](#)

COMMUNITIES

- [Standard Actions Overrides!](#)
- A few personalization updates with [multi-level nav bar](#) and [some standard component changes](#).
- Sharing data with external users now simpler thanks to [External Account Hierarchies](#)
- Salesforce have [Fixed a bug for the Service Appointment Component](#) which requires some manual setup.
- Other notes were either Previously announced (no more [Guest user owners](#), etc) or Pilot ([LWC Build-your-Own](#))

SERVICE

- Salesforce are releasing [their own CTI](#) which [seems to be a front for Amazon Connect](#) and some SF polish like [automated transcription](#). Comes at extra cost, unsure if worth it ATM.
- Bots get [Multi-Language Support](#), [Best-Practice Help Pages](#), [Skill-Based Routing to Agents](#), and [Builder](#) upgrades.
- Routing now allows [optional skills](#) which avoids huge queue times.
- Surveys are still not as good as GetFeedback but are [slowly getting there](#). [It’s kind of peppered accross the entire Release Notes though](#)

REPORTS AND DASHBOARDS

- [Flow Interviews](#) now also available for reporting
- [PriceBookEntries](#) now targetable when creating Custom Report Types
- This was already the case for Formatted Data, but now [CSV](#) can now also be attached to subscriptions

EINSTEIN

- I may have missed this before but it seems that Salesforce have now decided to cover themselves re: the use of the Einstein name: "[Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.](#)" which I find funny
- Einstein features have gotten a lot of love in the last year, and are getting at a point where I could start recommending clients buy them.
- [Prediction Builder now supports Field-to-Field Comparisons](#)
- [Case-WrapUp](#) is a nice addition to Service Einstein
- Einstein Search now gets [Natural Language Processing](#) and is no longer limited by [Shield Encryption](#)

DEVELOPMENT

- [Unlocked Packages](#) will require Code Coverage to Promote and can also be [Org-Dependent](#)
- [Einstein OCR via API](#)
- Sending [VF Emails to Guest Users](#) requires extra parameter, check if you're subject to this.
- [Reusable CSS](#) between LWCs. I think design tokens are still a bit wonky though.
- Use the `@salesforce/community` scoped module to import information about the [current community context](#)
- [Connect REST Api now returns SVG instead of Flash](#)
- [Generic SObject](#) data types are here and can be used in Flow actions. This means you can use `sObject` and `sObject[]` in your flow screen LWCs!
- [Private Connect](#) sets up a fully managed network connection between your Salesforce org and your AWS Virtual Private Cloud (VPC)
- [Crypto](#) now supports RSA-SHA384 and RSA-SHA512.
- [TOTP](#) via APEX now possibilities
- CDC Events now support [mandatory sent fields in events](#)
- Dev and Dev Pro Sandbox now support [scratch org type of source tracking](#). Hopefully they will support at least partial sandboxes in the future!

- `System.Security.stripInaccessible()` now [respect relationship fields](#). It's now minimal overhead to add FLS / CRUD checker to your apex. Finally.
- New [@JSON annotation](#) controls whether instances of the class can be serialized or deserialized. Default setting in API v48+ more restrictive than current situation.

DOGELAND

- [:doge:](#) Someone at SF thought it was a good idea to re-open the [Work.com casket](#) but put a Sales pitch instead of content, and brand it as COVID relief. Reception mixed at best.
 - [:doge:](#) Salesforce PRESENTS: [OPTIMIZER](#) ! Yes it's the same (awesome) tool that was first an external website and then a menu item. No, we don't know why it's in the release notes. [Again](#). And [Again](#). And [Again](#).
 - [:doge:](#) [In-App Guidance got an upgrade with Walkthroughs](#) buuuuuuuut you need MyTrailhead to use it. Too bad because it looks [Dope as Hell](#)
 - CMT team wins "weirdest Jargon" award for an otherwise [cool release note](#)
-

This abridged version was graciously written up by the **SF Discord**

We have a nice wiki: <https://wiki.sfxd.org/>

And a linkedin page: <https://www.linkedin.com/company/sfxd/>

Join the ~2000 members in the most active chat-based community around Salesforce these parts of the web at <http://join.sfxd.org/>

SPECIAL EDITION - FUTURE OF AUTOMATION

I put this outside of the ARN themselves because, well, it's not a release note.

Re: After/Before/Events Triggered Flows:

- HUGE SPECULATION as always - I'm calling Process Builder deprecated by 1y and fully retired by 2y. To the angry people that flamed me when I called Aura [deprecated](#): [BOOYAH](#).
- Using PB is still required due to current Flow limitations, and I expect that there will be a tool to migrate one to the other seeing as technically they're both the exact same tool in the backend - so don't panic about your Process Builders just yet.
- These changes add yet another automation source to Objects. That new automation source probably won't play super well with others, as the [Before Save](#) flows highlighted

with APEX.

- The new Flow execution contexts don't support everything that is in PB or even normal Flows yet (no Subflows, no `ISCHANGED()`).
- In Summary while these are very exciting times and highlight a bright future of Low-Code possibilities, we currently advise NOT to use these new flows outside of Prototyping or Sandboxes.

Re: User-Permission Bypassing Flows:

- While SF seems to push Flows as Admin-friendly, the reality is that as time goes on, Flows are becoming more of Procedural programming with a GUI than anything else.
- The above effectively puts the skill floor higher for any new admins - think 7 years ago when all you had was page layouts and workflows. Things are a bit more complex now.
- User-Permission Bypassing means you can effectively do VERY BAD THINGS, and to be honest I would rather they hide this option behind a few layers of prompts asking if you know WTF you're doing. As-is I can totally see compliance issues in the medium-term. I encourage any Auditors or Consulting partners to add checking for this permission in any audit checklist and ensure that it's needed.
- For all SF Admins reading this, SFXD already told you this last release, but really: Train yourself on Flows, Fast. You're going to need it.

Re: Dynamic Forms:

- **Dynamic Forms** will extend quite a lot what can be done in Lightning Page Layouts. It pretty much replaces Layouts in general.
- The current status is Preview, meaning SF don't want to support it just yet. If you're using it in Production, this is bad. Don't do it. Wait a few releases more.
- I am still unsure what this means for Page Layouts/Record Types in the future. Probably will be decorrelated and based on conditional display (which doesn't work well ATM)
- As-Is this is a very strong Do Not Use.
- It seems that more and more things are getting **Builders** in the backend which allow more powerful stuff in the hands of admins (Flows, Bots, Paths). While this does sound like great news, it'll raise the skill floor to be a competent Salesforce Manager in the next years

*Re: Lightning Message Service (**LMS**) and continued support of legacy code:*

- With **LMS** there is now one ring to rule them all. In its most ideal usage, this now serves as the primary inter-component communication mechanism **between** LWCs that are in separate component trees.
 - But, until your org(s) reach the state where you can invest 100% in LWCs only (because you are moving to that future, right?) this is now a quick win for bringing your VF and Aura solutions to play nicely with your shiny new LWCs.
 - Now that they have finally settled on conventions (**LMS** using various **channels**), I fully expect this to be the backbone for some cool event driven application hooks in the future.
-

Revision #14

Created 22 May 2020 13:56:22 by Windyo

Updated 28 May 2025 10:26:04 by Windyo