

welcome to

dreamOle



Ebury

provar
Test Automation for Salesforce®

wefox
G R O U P

NUBIKA

Cloud
Coachers

Sharin Pix

s4g
Consulting

omega

FINANCIALFORCE

CLOUD
GALACTICOS
SALESFORCE SUPERSTARS

LIGHTNING
CHAMPIONS

AutoRABIT
DevOps Made Easy

QualityClouds
Your Favorite Clouds - Amazon AWS - Salesforce CloudWatch

Cognizant

#dreamOle19

Málaga



2nd Generation Packaging

Esteve Graells
MVP Architect
forcegraells.com

Málaga

Releasing *is* choosing your menu (paradigms)



Metadata Happy Soup deployed via tool

(+ Helping Tools:
Change Sets,
ANT, CLI
and 3rd Party like Copado,
Autorabbit)



Locked Packages

Because both have things that I



So, wouldn't be great to have a
third main course with the best
of both?

- Keeping my **knowledge** and the **flexibility** of the Happy Soup
- But using the “**packaging paradigm**” used on Locked Packages
- But **avoiding** the **effort** required for creating/testing/deploying for Locked packages because I am not an ISV

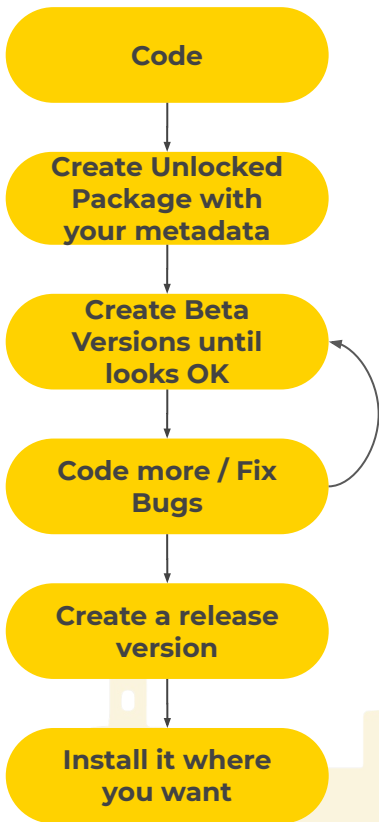


Unlocked Packages is our special today



1. **Break and Organize** your **metadata** in *your manner*
2. **Version packages** at your own pace in a **controlled** process
3. **Inter-dependencies supported** with other unlocked packages, classic metadata and packages
4. **100% fully automated capable** and integrated on the SFDX CLI
5. **Incremental adoption** for your projects (starting small)
6. **GA Winter' 19**

The process is easy: let's see it in action



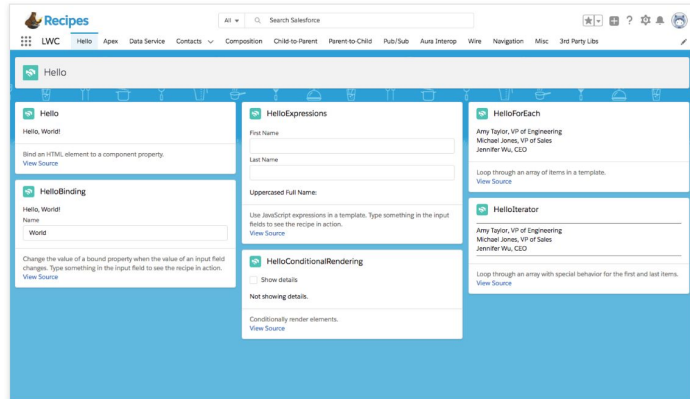
- **Code / Download / get your code**
- **Create a Unlocked** package (In your org oy using the CLI)
- **Create a new package version** (it becomes Beta)
 - You can install it in non-Prod Orgs by CLI or by Link)
- **Install it in a non-Prod Org**
- **Change your code**
 - Use DX Development Lifecycle with (push, pull, load data, etc.)
- **Iterate 3-5**
 - Automated preferably
 - Maintaining Version Name and Version Number
- **Create a Release version** Package
- **Install it in a Production Org** by CLI or by using a Link (remember you can install any ORG you need)

Get source / create source



Recipes

⚡ Lightning Web Components



Description

A collection of easy-to-digest code examples for Lightning Web Components. Each recipe demonstrates how to code a specific task in about 30 lines of code or less. A View Source link takes you right to the code in GitHub. From Hello World to data access and third-party libraries, there is a recipe for that!

Features

- Lightning Web Components
- Lightning Data Service
- Composition
- Events
- Pub/Sub
- @wire
- Navigation
- 3rd Party Libraries

git clone <https://github.com/trailheadapps/lwc-recipes>
Cloning into 'lwc-recipes'...

Set up a Dev Hub Org



```
{
  "packageDirectories": [
    {
      "path": "force-app",
      "default": true
    }
  ],
  "namespace": "",
  "langauge" : "en_US", //I want all my scratchs in english
  "sfdcLoginUrl": "https://login.salesforce.com",
  "sourceApiVersion": "45.0"
}
```

A default sfdx-project.json



Create the Package



```
sfdx force:package:create --name recipes-dreamole --description "Recipes for dreamole" --packagetype Unlocked --path force-app --nonamespace --targetdevhubusername DevHub
```

sfdx-project.json has been updated. Successfully created a package. 0Ho1r000000fxTJCAY

=== Ids

| NAME | VALUE |
|------|-------|
|------|-------|

| | |
|------------|--------------------|
| Package Id | 0Ho1r000000fxTJCAY |
|------------|--------------------|

Name, description, type and the Hub, that's all!!

The CLI automatically updates the sfdx-project.json

```
{
  "packageDirectories": [
    {
      ...
      "package": "recipes-dreamole",
      "versionName": "ver 0.1",
      "versionNumber": "0.1.0.NEXT"
    }
  ],
  ...
  "packageAliases": {
    "recipes-dreamole": "0Ho1r000000fxTJCAY"
  }
}
```

You can create Objects with the Package Manager and Dependencies will be detected

The screenshot shows the Salesforce Package Manager interface. At the top, there's a 'SETUP' button and the 'Package Manager' title. Below that, the package name 'DreamOlé Base Objects' is displayed with a link to 'Back to Package List'. The 'Package Detail' section includes buttons for 'Edit', 'Delete', and 'Upload'. A table shows package information: Package Name (DreamOlé Base Objects), Language (English), Type (Unmanaged), and a 'Notify on Apex Error' checkbox. Below this, a table shows 'Created By' (Esteve Graells) and 'Last Modified By' (Esteve Graells) with timestamps. The 'Components' tab is active, showing a table with columns: Action, Name, Parent Object, Type, Included By, and Owned By. One component is listed: 'ChangePasswordController' (Apex Class) with 'User Selected' as the included by and owner.

| Package Name | DreamOlé Base Objects | Type | Unmanaged |
|----------------------|----------------------------------|------------------|----------------------------------|
| Language | English | | |
| Notify on Apex Error | | | |
| Created By | Esteve Graells, 12/04/2019 10:24 | Last Modified By | Esteve Graells, 12/04/2019 10:24 |
| Description | | | |

| Action | Name | Parent Object | Type | Included By | Owned By |
|------------------------|--------------------------|---------------|------------|---------------|----------|
| Remove | ChangePasswordController | | Apex Class | User Selected | |

Create a Scratch Org to Test the Package



```
sfdx force:org:create --definitionfile config/project-scratch-def.json --setalias TestingDreamolePckg -v DevHub
```

Successfully created scratch org: 00D9E000000111hUAA, username: test-x12xurol2axp@example.com

```
sfdx force:org:list
```

```
=== Orgs
```

| ALIAS | USERNAME | ORG ID | CONNECTED STATUS |
|-------|----------|--------|------------------|
|-------|----------|--------|------------------|

| | | | |
|------------|---------------------------|--------------------|-----------|
| (D) DevHub | esteve.graells2@gmail.com | 00D1r000000t7BVEAY | Connected |
|------------|---------------------------|--------------------|-----------|

| ALIAS | SCRATCH ORG NAME | USERNAME | ORG ID | EXPIRATION DATE |
|-------|------------------|----------|--------|-----------------|
|-------|------------------|----------|--------|-----------------|

| | | | | |
|---------------------|-------------|-------------------------------|--------------------|------------|
| MyScratchOrg | Dreamhouse | test-cds4oepspq1o@example.com | 00D0E000000DgymUAC | 2019-05-08 |
| TestingDreamolePckg | LWC Recipes | test-x12xurol2axp@example.com | 00D9E000000111hUAA | 2019-04-16 |
| darrera | GIFter | test-bxoxvwgvyz31@example.com | 00D0E000000Dk7hUAC | 2019-04-15 |

Could be a sandbox as well!!

Create a Beta Version for the package



```
sfdx force:package:version:create -p recipes-dreamole -d force-app -k secretkey --wait 10 -v DevHub
Request in progress. Sleeping 30 seconds. Will wait a total of 600 more seconds before timing out. Current
Status='Queued'
```

...

Successfully created the package version [08c1r000000fxUHAAY]. Subscriber Package Version Id:
04t1r000000Ah9TAAS

Package Installation URL: <https://login.salesforce.com/packaging/installPackage.apexp?p0=04t1r000000Ah9TAAS>

As an alternative, you can use the "**sfdx force:package:install**" command.

```
"packageAliases": {
  "recipes-dreamole": "0Ho1r000000fxTJCAY",
  "recipes-dreamole@0.1.0-1": "04t1r000000Ah9TAAS"
}
```

The CLI, again, automatically updates
the sfdx-project.json

Install it In the Scratch Org created



```
sfdx force:package:install --wait 10 --publishwait 10 --package recipes-dreamole@0.1.0-1 -k  
secretkey -r -u TestingDreamolePckg
```

Waiting for the package install request to complete. Status = IN_PROGRESS

...

Successfully installed package [04t1r000000Ah9TAAS]

```
sfdx force:org:open -u TestingDreamolePckg
```

```
sfdx force:org:open -p lightning/n/Hello -u DevHub (open a Tab directly)
```

Change your code, add/remove components, dependencies, pull/push code via CLI
... iteratively.



Deploy to PROD (1): Create a Release Version

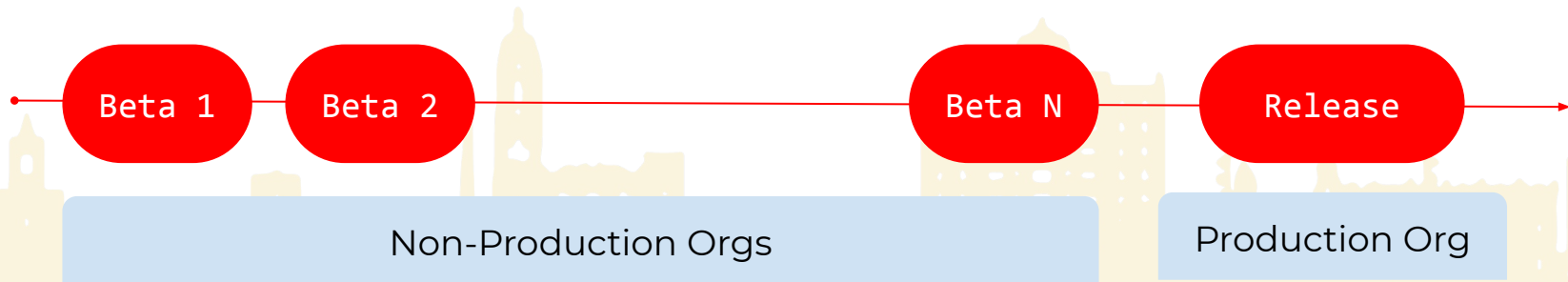


```
sfdx force:package:version:promote -p recipes-dreamole@0.1.0-1 -v DevHub
```

```
Are you sure you want to release package version recipes-dreamole@0.1.0-1? You can't undo this action. Release package (y/n)? y
```











Successfully promoted the package version, ID: 04t1r000000Ah9TAAS, to released.

You can install beta package versions in scratch orgs, sandboxes, and a released package version in any org, but only a released one in Production Orgs.



Waiting for the package install request to complete. Status = IN_PROGRESS

Successfully installed package [04t1r000000Ah9TAAS]

| Action | Package Name | Publisher | Version Number | Namespace Prefix | Install Date | Limits | Apps | Tabs | Objects | AppExchange Ready |
|---|---|--------------------------------|----------------|------------------|------------------|---|------|------|---------|-------------------|
| Uninstall |  Salesforce Connected Apps | Salesforce.com | 1.7 | sf_com_apps | 10/03/2018 19:03 |  | 0 | 0 | 0 | Not Passed |
| <div>Description</div> <div>This package contains Connected Applications for all the officially supported Salesforce client applications such as Touch, Salesforce for Outlook, Sa...</div> | | | | | | | | | | |
| Uninstall |  Slack | Slack Essentials Packaging Org | 1.7 | slack_essential | 19/09/2018 18:19 |  | 0 | 0 | 2 | Passed |
| Uninstall |  GIFter | Personal | 1.1 | | 8/04/2019 18:17 |  | 1 | 1 | 0 | Not Applicable |
| <div>Description</div> <div>GIPHY on Chatter</div> | | | | | | | | | | |
| Uninstall |  recipes-dreamole | Personal | 0.1 | | 9/04/2019 12:12 |  | 1 | 12 | 0 | Not Applicable |
| <div>Description</div> <div>Recipes for dreamole</div> | | | | | | | | | | |
| Uninstall |  th-dreamhouse | Personal | 1.0 | | 8/04/2019 15:35 |  | 1 | 7 | 5 | Not Applicable |
| <div>Description</div> <div>My Dreamhouse Package</div> | | | | | | | | | | |

Things to bear in mind

Before call to arms

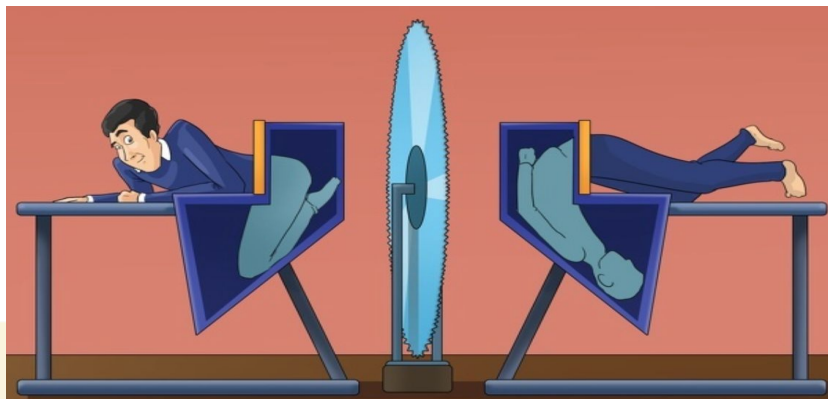


Check before adopting Unlocked Packages



IMHO the **key** to use Unlocked Packages is not the technology, but how to **break/organize** our current happy soup.

1. **Manage Dependencies** will be hard at the beginning.
2. Unlocked Packages **are easy-peasy to automate**, but a tested branching schema should be present
3. **Metadata Coverage** for Unlocked Packages is slightly different.

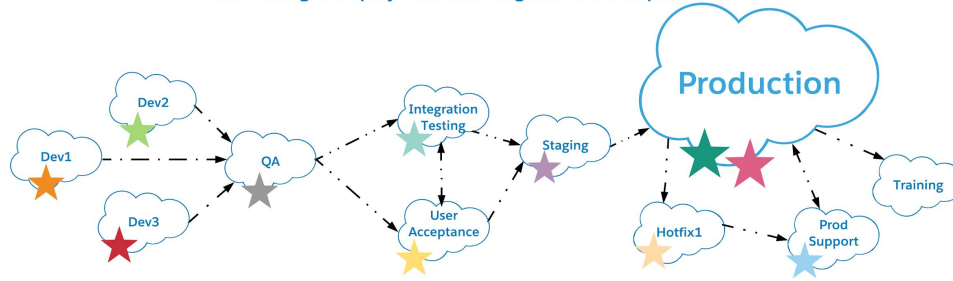


Check before adopting Unlocked Packages

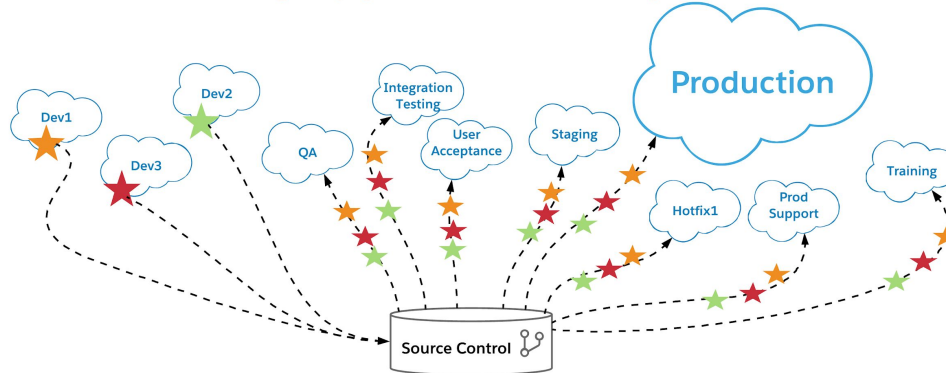


★ = Group of metadata

Non-Packaged Deployments and Org-Based Development



Packaged Deployments and Source-Based Development



Know more - practice even more!



- **Imagine a new source of truth (explanation of a new packaging method for customers)**
 - https://trailhead.salesforce.com/content/learn/modules/sfdx_dev_model/sfdx_dev_model_neworganization
- **Unlocked Packages for Customers:**
 - <https://trailhead.salesforce.com/content/learn/modules/unlocked-packages-for-customers>
- **Quick Start: Unlocked Packages**
 - <https://trailhead.salesforce.com/content/learn/projects/quick-start-unlocked-packages>
- **Working with Modular Development and Unlocked Packages Series by Zayne Turner (Lead Developer Evangelist)**
 - <https://developer.salesforce.com/blogs/2018/06/working-with-modular-development-and-unlocked-packages-part-1.html>
- **(Video) How Everyone Can Leverage Salesforce DX Packaging**
 - [How Everyone Can Leverage Salesforce DX Packaging](#)
- **Developer docs - Unlocked Packages (Generally Available) and Second-Generation Managed Packages (Beta)**
 - https://developer.salesforce.com/docs/atlas.en-us.218.0.sfdx_dev.meta/sfdx_dev/sfdx_dev_dev2gp.htm
- **Easy Spaces is an app that focuses on the modularization with unlocked packages**
 - <https://github.com/trailheadapps/easy-spaces>
- **Packaging 2 Beta Trailblazer Community**
 - https://success.salesforce.com/_ui/core/chatter/groups/GroupProfilePage?q=0F93A000000Lg5U
- **Package Development Readiness**
 - <https://trailhead.salesforce.com/en/content/learn/modules/package-development-readiness>





Thank you

#dreamOle19

Málaga

dreamOle

Q&A



Ebury

provar
Test Automation for Salesforce®

wefox
G R O U P

NUBIKA

Cloud
Coachers

Sharin Pix

s4g
Consulting

omega^{cm}

FINANCIALFORCE

CLOUD
GALACTICOS
SALESFORCE SUPPLYTANK

LIGHTNING
CHAMPIONS

AutoRABIT
DevOps Made Easy

QualityClouds
Cloud Platform Experts - Amazon AWS - Cloud Platform Specialists

Cognizant

#dreamOle19

Málaga