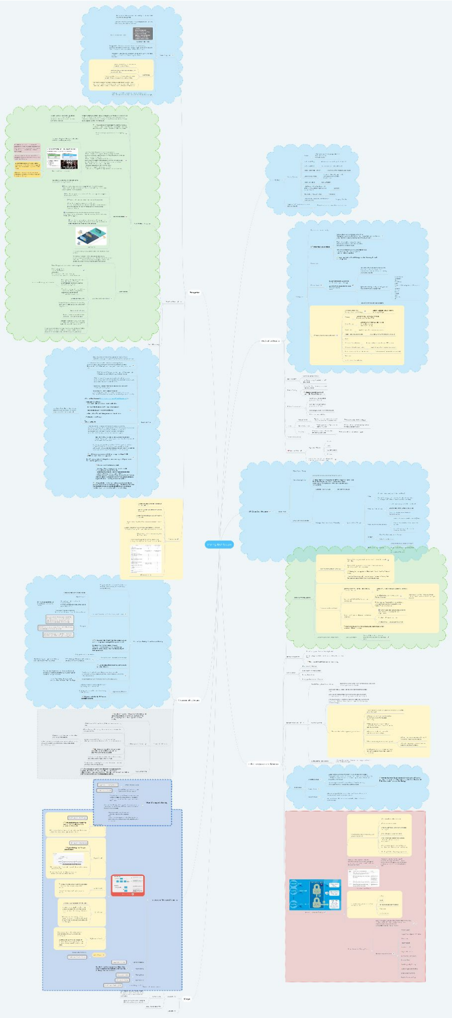
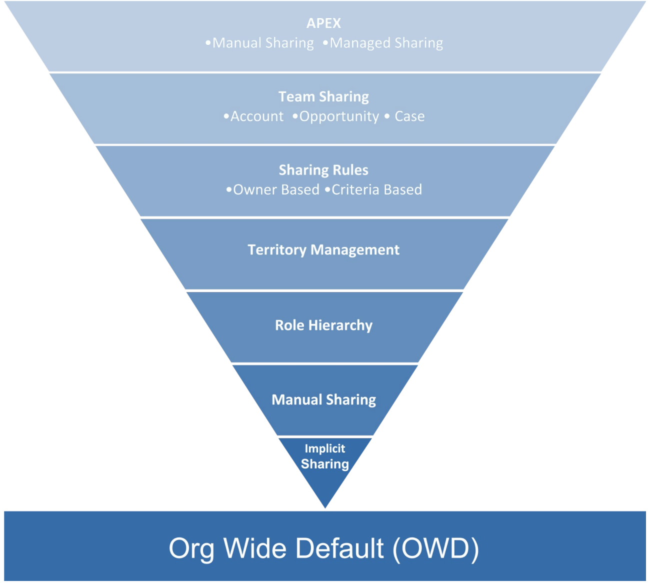
Account Hierarchy Sharing and ETM Planning with Salesforce Maps

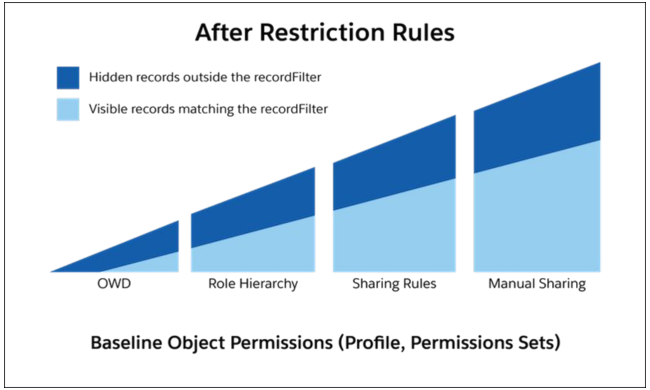
Svet Voloshin

Salesforce Security Model

Mindmap →
(click to download)

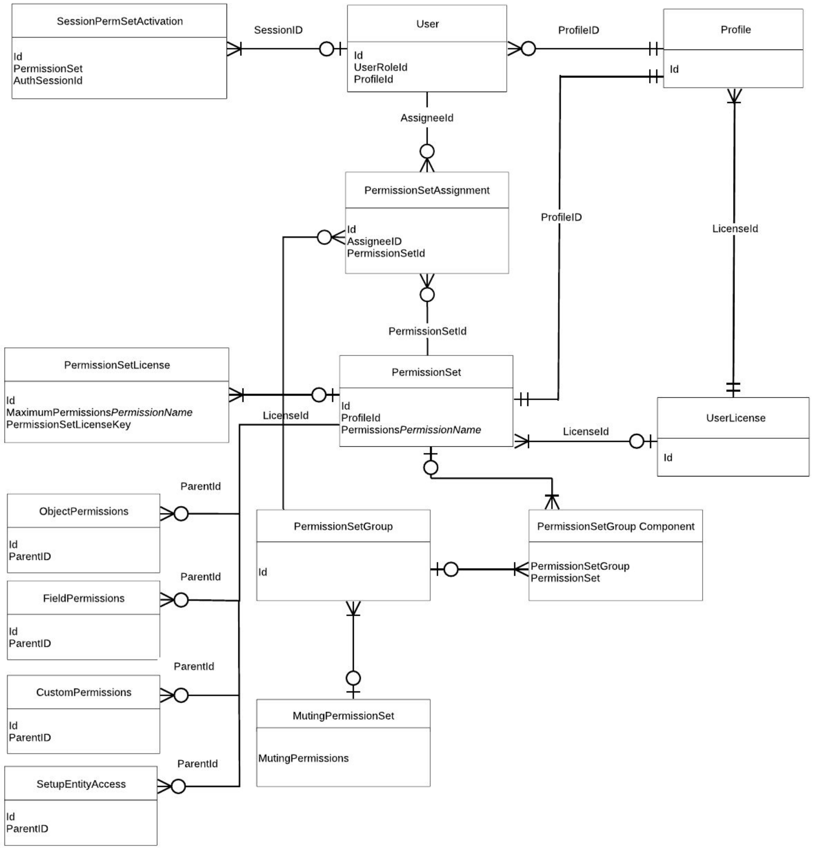






Profiles and Permission Sets

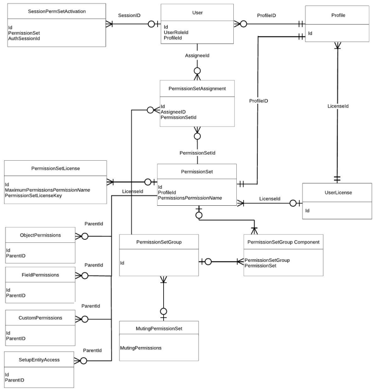
- Best Practice is to use Permission sets to manage permissions.
- Profiles should be broken out for other requirements like page layout assignments or sharing sets.
- Salesforce offers Permission Set Groups and Muting Permission Sets.



Muting Permission Sets

What is a good use case for a ***muting*** perm set?

- Muting perm sets apply to perm set groups, not to perm sets themselves.
- When upgrading a massive managed package, like Copado. Some managed packages come with a bundled perm set, which users clone and adjust.
- Muting perm sets make it easier to keep track of disabled permissions and additional custom perm sets extend functionality as designed.
- User ends up with a Perm Set Group, consisting of:
    - Managed Package Permission Set
    - Additional Permission Set
    - Muting Permission Set



Account Hierarchy

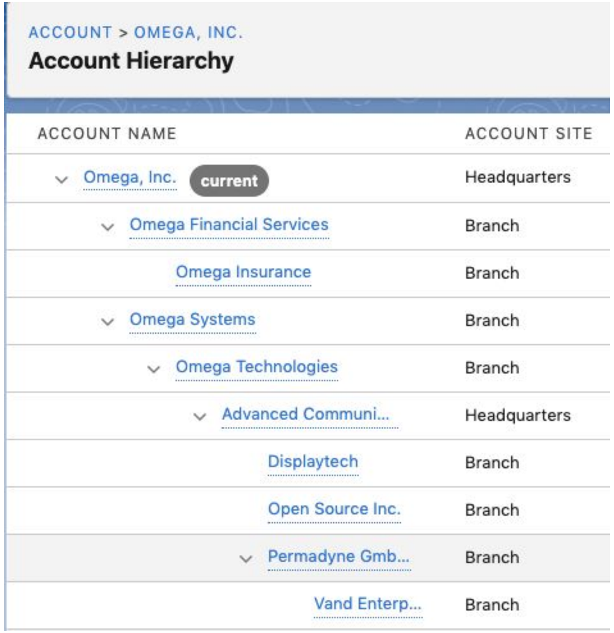
NOT to be confused with the External Account Hierarchies

Account Hierarchies' Impact on Data Access

A lot of people make a bad assumption when they implement an account hierarchy. They assume the users who can access a parent account can also access the children accounts. The simple fact of only having a parent/child relationship between two records doesn't drive access. Although the role hierarchy and the territory hierarchy do work in this way, the account hierarchy doesn't.

*Source: Salesforce - **A Guide to Sharing Architecture (Winter '23)***

Set Up Account Hierarchy in Lightning Experience

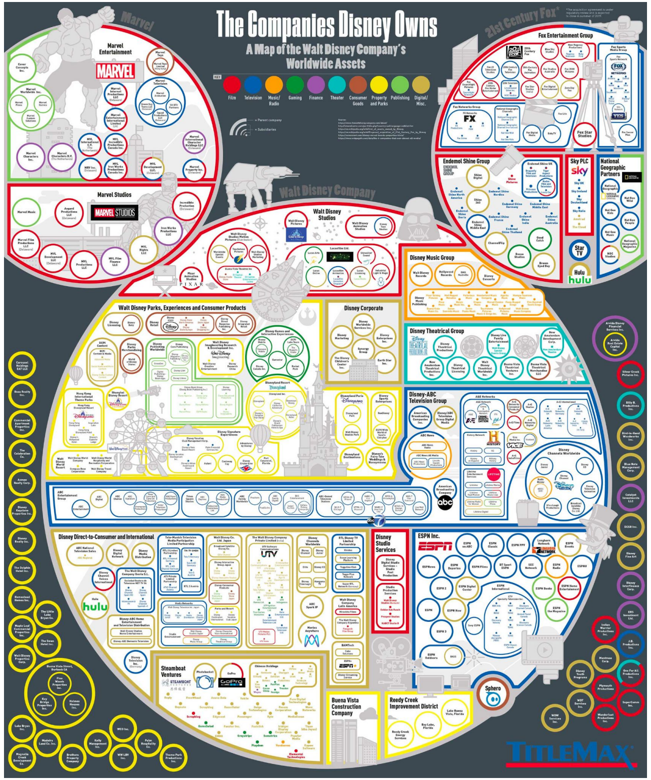


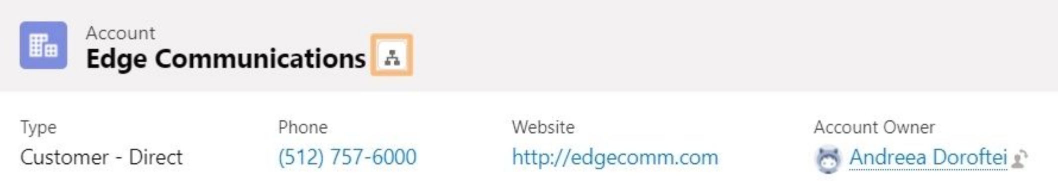
Account Hierarchy Sharing - Continued…

What if we could make it into a data sharing mechanism? Dare to dream!

What is it for?

- The Salesforce Account Hierarchy feature shows how Account records relate to one another, to represent parent companies and their subsidiaries (each represented by an Account). Accounts can be organized in a layered way as they are grouped based on criteria (e.g. Country, State, or perhaps by division).





Account Hierarchy - Ultimate Parent

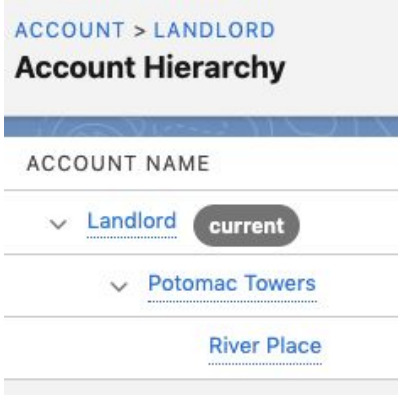
Why make Account Hierarchy into a data sharing mechanism?

- External Account Hierarchies exist for sharing purposes, but why not *Internal*?
    - There can be many business reasons - use your imagination...
- How to do it?
    1. Create an Ultimate Parent Formula Field
        a. User-friendly Ultimate Parent field will show the **name** of the Account, but that would not help for sharing (hint: use the same formula, but substitute “Id” for “Name”.
        b. For sharing to work, you would need a Record Id, which you can use in APEX sharing or even in a Flow.

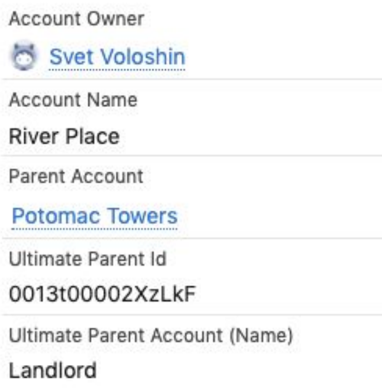
Account Hierarchy Sharing

How to share the Ultimate Parent record:

1. Get the Ultimate Parent Id from the child record and create a *.share* record for it.
2. Query <u>all</u> account records with the same Ultimate Parent record and create *.share* records for it.



ParentId	The Id of the record being shared. This field cannot be updated.
UserOrGroupId	The Id of the User to whom you are granting access. May also be a Public Group Id, Role Id, or Territory Id. This field cannot be updated.
AccessLevel	The level of access that the specified User or Group has been granted. Valid values for Apex managed sharing are: Edit, Read. This field must be set to an access level that is higher than the organization's default access level for the parent object. For more information, see [Access Levels.]()
RowCause (aka Sharing Reasons)	The reason why the user or group is being granted access. The reason determines the type of sharing, which in turn controls who can alter the sharing record. This field cannot be updated.



Account Hierarchy Sharing - Considerations

- Share table available **only** when Organization Wide Default [OWD] sharing is **not** Public Read Write
- Apex sharing reasons can be created **only for custom objects**
- As Apex Sharing Reason not available for Standard objects only way to create Apex based sharing for Standard objects are using row cause **Manual**
- As row cause is **Manual** for **Standard objects**, Apex based sharing would be **lost** once owner is changed [Because its behavior of Manual Sharing]
- In case of **custom** object & custom Apex Sharing Reason, sharing **would not be lost** when owner changed
- It’s not possible to trigger on ***.share*** object/record - what are the options?
    - Use a scheduled job/Flow
    - Create a custom button
    - Create an LWC

Account Hierarchy Sharing - Pros & Cons

Pros

- Makes it faster to share the entire account hierarchy with a user or a group of users
- Interesting exercise if you're a Salesforce nerd :)

Cons

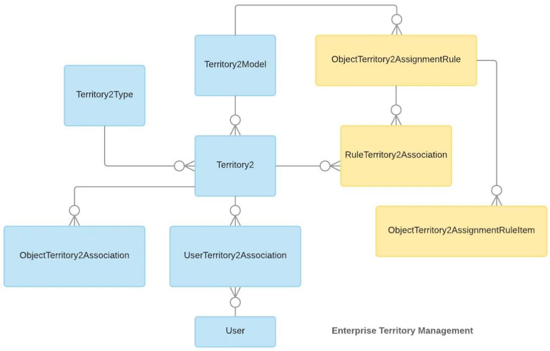
- Very hard to make this scalable
- Potential for data leakage
- More development for *unsharing*
- Is it truly necessary to share the entire hierarchy?

Implicit Sharing

Type of Sharing	Provides	Details
Parent	Read-only access to the parent account for a user with access to a child record	Not used when sharing on the child is controlled by its parent Expensive to maintain with many account children When a user loses access to a child, Salesforce needs to check all other children to see if it can delete the implicit parent.
Child	Access to child records for the owner of the parent account	Not used when sharing on the child is controlled by its parent Controlled by child access settings for the account owner's role Supports account sharing rules that grant child record access Supports account team access based on team settings When a user loses access to the parent, Salesforce needs to remove all the implicit children for that user.
Site or Portal	Access to a site or portal account and all associated contacts for all site or portal users under that account	Shared to the lowest role under the site or portal account
High Volume	Access to data owned by high-volume users associated with a sharing set for users member of the sharing set's access group	All members of the sharing set access group gain access to every record owned by every high-volume user associated with that sharing set
High Volume Parent	Read only access to the parent account of records shared through a sharing set's access group for users member of the group	Maintains the ability to see the parent account when users are given access to account children owned by high-volume users

Enterprise Territory Management - Fast Facts

Territory management is an **account sharing system** that grants access to accounts based on the characteristics of the accounts. It enables your company to structure your Salesforce data and users in the same way as you structure your sales territories.



- Forecasts are supported
- Both Territory management sharing and role hierarchy sharing can be active at the same time.
- Users will receive whatever access is most permissive across both hierarchies
- Territories, as well as Territories and Subordinates can be members of Public Groups
- Territories, as well as Territories and Subordinates can be members of Sharing Rules
- ETM can be disabled (new)
- Only 1 <u>Active</u> Territory is allowed
- Unlimited Edition can have up to 4 territories in Planning state (2 for Enterprise)
- OpportunityTerritory2AssignmentFilter Global Interface - default APEX Class that comes out of the box used to assign a single territory to an Opportunity
    - Accounts can belong to multiple territories
    - Opportunities can belong to only a single territory

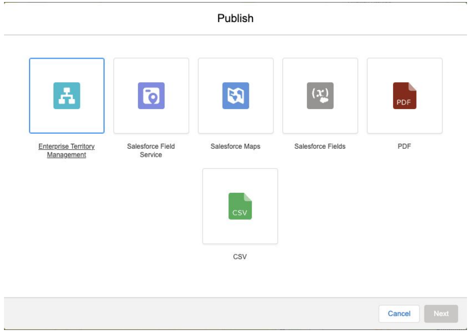
Enterprise Territory Management - Limits

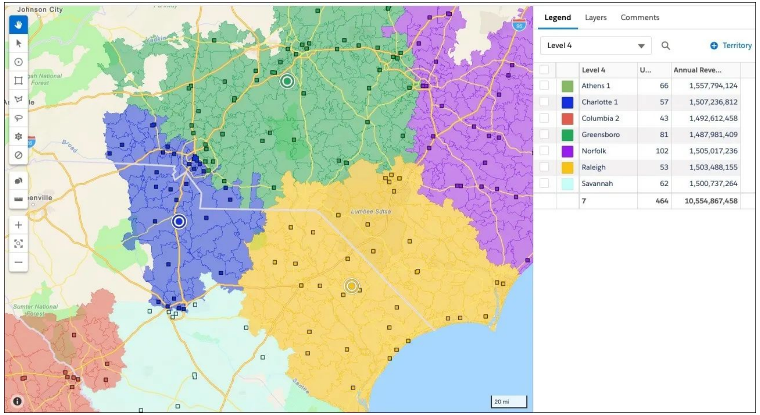
- By default, each territory model can have up to 1,000 territories.
- you can add up to 99,999 territories to a territory model by contacting Salesforce Customer Support. Requests for more than 20,000 territories per model are subject to approval.
- Up to 15 assignment rules per territory
    - Theoretical limit: 1,499,985 assignment rules

Territory Planning - Salesforce Maps

- Territory Model planning can take months - it's not uncommon to do this quarterly
- One can use **Salesforce Maps** to do the territory planning by visualizing geographical regions
  - Datasets are created
  - Alignment Details are created using Datasets (See Screenshot)
- Once you're done planning territories in Salesforce Maps, they are published as normal territory models and are activated.
- You can also publish to [Field Service Service Territories]

(Note: When specifying publish new, duplicate territories in Territory Planning or Field Service will cause a failure, as Field Service does not allow for duplicate names.)

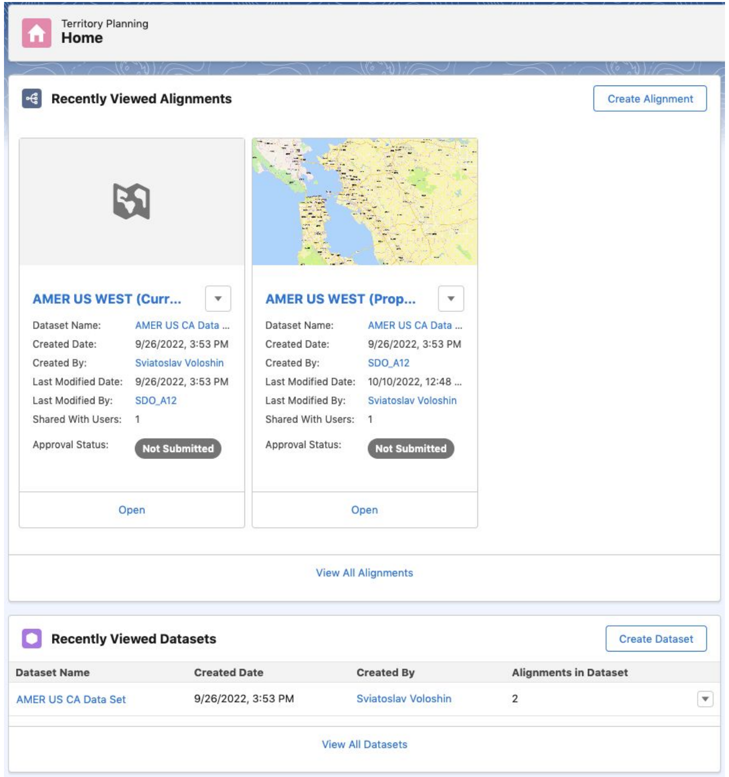




Trailhead: [Get Started with Territory Planning]

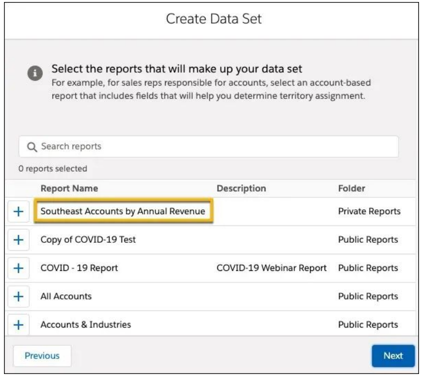
SF Maps Territory Planning Features & Functionality

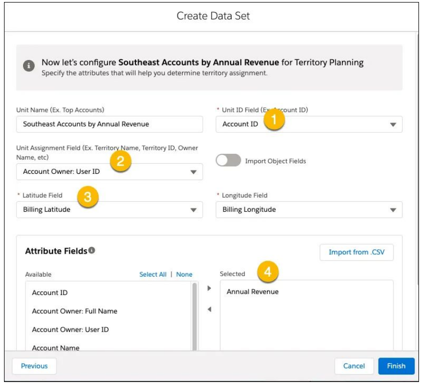
Territory Planning Features	Functionality
Access Advanced Territory Management and Optimization	• Build and implement optimized territory alignments. • Maximize geographic coverage without affecting current assignments.
Import optimized territories into Salesforce Maps	• View territories geographically. • Plan travel by using drive-time and road networks for optimal routes and schedules.
Manage territories inside Salesforce	• Visualize complete territory designs. • Manage and communicate territory configurations with leaders and field reps.



SF Maps Territory Planning Steps

1. Create a Report
    a. Object ID field
    b. Fields used to assign territories, like User Name, User ID, Territory Name, or Territory ID
    c. Latitude field*
    d. Longitude field*
        i. *Geocoded records are required.
2. Plan the Design (Trailhead Simulator)
    a. **Data Sets**: A collection of records that define your Territory Alignments.
    b. **Territory Alignment**: The proposed territory model for the group of records defined in the Data Set. An alignment provides the visual framework of your territory classifications. For example, territories may be classified as regions or districts.
3. Create an Alignment (Trailhead Simulator)
    a. By sales rep or service technician
    b. By size of business, such as mid-market and small business
    c. By managers assigned to different teams





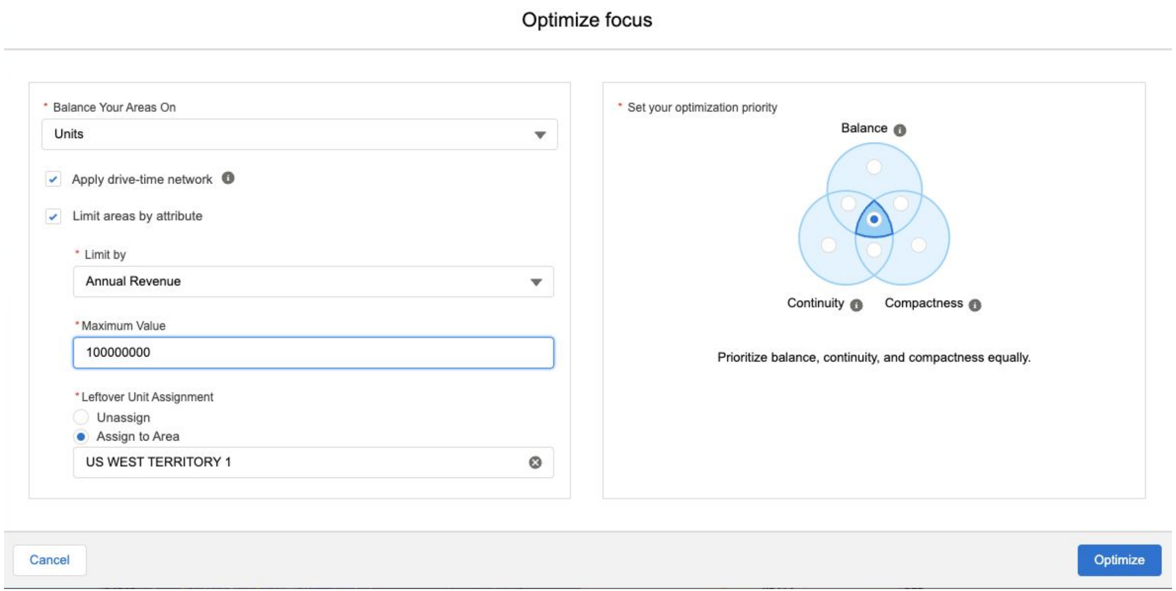
Territory Alignment Optimization

Business Trend: *due to the changed nature of selling in the “post-COVID” era, a lot of selling is happening remotely.*

- There can be a hybrid mix of high-value accounts concentrated within a driving distance of one-another, while accounts of lesser value/priority can be reached remote

Territory Alignments can be optimized based on:

- **Balance**: equal distribution of the selected attribute across Territory areas.
- **Continuity**: Minimum amount of Unit/COntainer reassignments across Territory areas.
- **Compactness**: minimum distance between Units and Containers within Territory Areas



Enterprise Territory Management - Resources

- Salesforce Maps Territory Planning Demo
- Webinar: How Salesforce Maps & Territory Planning Drive Revenue and Sales Efficiency (skip to 23:00-minute mark)
- Automating with Territory Management (declarative Flow)
- Enterprise Territory Management – Auto Account Assignment using Apex (Jitendra Zaa)
- ETM Implementation - Official Implementation Guide
- Salesforce Enterprise Territory Management Cheat Sheet
- Trailhead: Salesforce Maps Territory Planning
- Salesforce Maps Pricing
    - Salesforce Maps Territory Planning
    - $75 user/month*
    - Get advanced optimization that balances territories and takes plans out of spreadsheets.

Thank you!

Stay tuned in the Slack channel for the next topic and please feel free to suggest areas of interest.

Ways to get in touch...

Connect with me on LinkedIn
Email: svet@dc3me.com

Obvious People Slack Channel:
#salesforce-academy