

Opportunities Hierarchy

Account Name		Account Type		Industry		Annual Revenue
United Oil & Gas Corp.		Customer - Direct		Energy		\$5,600,000,000.00
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private	
1 United Oil Installations	Negotiation/Review	\$270,000.00	11/26/2019	Yet to begin	✓	
2 United Oil Refinery Generators	Closed Won	\$915,000.00	1/7/2020	Completed		
3 United Oil Installations	Closed Won	\$235,000.00	12/7/2019	In progress		
4 United Oil Emergency Generators	Closed Won	\$440,000.00	11/19/2019	Completed		
5 United Oil Standby Generators	Closed Won	\$120,000.00	1/22/2020		✓	
6 United Oil Plant Standby Generators	Needs Analysis	\$675,000.00	12/24/2019			

Account Name		Account Type		Industry		Annual Revenue
United Oil & Gas, Singapore		Customer - Direct		Energy		\$1,780,000,000.00
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private	
1 United Oil Office Portable Generators	Negotiation/Review	\$125,000.00	11/30/2019			
2 United Oil Refinery Generators	Proposal/Price Quote	\$270,000.00	1/14/2020	Yet to begin		
3 United Oil SLA	Closed Won	\$120,000.00	1/21/2020	In progress		

Account Name		Account Type		Industry		Annual Revenue
United Oil & Gas, UK		Customer - Direct		Energy		\$240,000,000.00
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private	
1 United Oil Installations	Closed Won	\$270,000.00	11/25/2019	In progress		

COMBINED HIERARCHY

by ARES

USER GUIDE

v1.6.0

INDEX

[Version History](#)

[Overview](#)

[Installation](#)

[Component Configuration](#)

[Component Attributes](#)

[Sample configuration](#)

[Known Issues](#)

[About ARES](#)

VERSION HISTORY

Version 1.1 - Initial release September 2020

Version 1.2 - Expand/collapse option

Version 1.3 - Performance improvements

Version 1.4 - Added filters to hierarchy/tables

Version 1.5 - Added option to use a fixed record Id and related record hierarchy

Version 1.6 - Added support for all custom objects

OVERVIEW

The Combined Hierarchy component is a reusable Lightning Component for Salesforce Lightning Experience to show a hierarchy of objects and their related records in the same view.

The Combined Hierarchy component allows the hierarchical relationships of your objects (standard and custom) to be displayed in a collapsible tree that includes related records in a table that can be sorted by any field.

The Combined Hierarchy component can be adapted for any standard or custom object that has a hierarchical relationship to itself. It is easy to use, just drag and drop the component on the Lightning page you want (Record, App, Community) and configure the fields to be shown in the hierarchy tree and in the table. That's all you need to do.

FLEXIBLE

- Fully customizable component: select any object types (including custom objects), select the hierarchical relationships, select relevant fields. Configure It to match your business scenarios.
- Use it wherever you want: in a record page, in a generic app page, in a community page, set fixed record id or get the record from the page context.
- See the big picture: reuse this component with different configurations in the same Lightning page to summarize information.

FAST

- Built using Lightning Web Components (a new iteration of the Lightning components to improve speed and interoperability).
- Lazy loading in tables to improve performance and speed (records are obtained on demand).

MODERN

- 100% built using Lightning Web Components, the new framework to follow web standards.
- Follows Lightning Design System guidelines: clarity, efficiency, consistency, beauty.
- Fully integrated in Salesforce: using standard Lightning Components and Lightning Design System style.

SECURE

- Field-level security and Create-Read-Update-Delete access is enforced, sharing settings allows to control who sees what.
- 100% native Salesforce solution: no external services callouts, your data never exits Salesforce.

INSTALLATION

Install from AppExchange: [here](#)

Once installed, “Combined Hierarchy (ARES)” component will be available in App Builder.

Remember to assign the Permission Set “Combined Hierarchy” to the users / profiles you want to use the solution.

Open Lightning App Builder and search for the component in the last section of the Lightning Components toolbar.

COMPONENT CONFIGURATION

COMPONENT ATTRIBUTES

Component attributes are divided in two categories:

Hierarchy Object attributes

These attributes control what object is used to display the hierarchy and what information is used in the different tree items of the hierarchy. Attribute API name is shown in parenthesis in case you want to use/extend the component in a custom page.

- ❖ ***Record Id (recordId)***
identifier of the record to get the hierarchy from
- ❖ ***Fixed Record Id (fixedRecordId)***
fixed value for record id in case you don't want to use binded value from page
- ❖ ***Related Record Field (relatedRecordFieldName)***
the field name to get the records in the hierarchy, in case it's not the current record on the page
- ❖ ***Hierarchy Object Name (treeObjectName)***
API name of desired object to be displayed in the tree
- ❖ ***Hierarchy Object Parent Field (treeParentFieldName)***
API name of the self-lookup field for the hierarchy object
- ❖ ***Hierarchy Object Main Field (treeMainField)***
API name of the field used as first column in the tree items (it will link to the record's page)

- ❖ **Hierarchy Object Fields (treeItemFieldsString)**
a comma-separated list of API fields names that you want to see as columns in the tree items
- ❖ **Hierarchy Object Filter (treeItemFilter)**
a valid where condition to filter tree items in the hierarchy. The current record (according to the recordId) will always be displayed

Table Object attributes

These attributes control what object is used to display the related table and what information is used as columns. Attribute API name is shown in parenthesis in case you want to use/extend the component in a custom page.

- ❖ **Table Object Name (tableObjectName)**
API name of desired object to be displayed in the tables
- ❖ **Table Object Parent Field (tableParentFieldName)**
API name of the lookup field that link table objects with hierarchy objects
- ❖ **Table Object Main Field (tableMainField)**
API name of the field used as first column in the tables (it will link to the record's page)
- ❖ **Table Object Fields (tableFieldsString)**
a comma-separated list of API fields names that you want to see as columns in the table
- ❖ **Table Object Filter (tableFilter)**
a valid where condition to filter elements of the table

Style attributes

These attributes control the component's style. Attribute API name is shown in parenthesis in case you want to use/extend the component in a custom page.

- ❖ **Card Icon (cardIcon)**
the icon name (according Lightning Design System nomenclature) to be shown in the header of the component.
See <https://www.lightningdesignsystem.com/icons/> for a full list of icons.
- ❖ **Card Title (cardTitle)**
the title to be shown in the header of the component

SAMPLE CONFIGURATION

Opportunity Hierarchy

Opportunities Hierarchy

Account Name United Oil & Gas Corp.		Account Type Customer - Direct		Industry Energy		Annual Revenue \$5,600,000,000.00	
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private		
1 United Oil Installations	Negotiation/Review	\$270,000.00	11/26/2019	Yet to begin	✓		
2 United Oil Refinery Generators	Closed Won	\$915,000.00	1/7/2020	Completed			
3 United Oil Installations	Closed Won	\$235,000.00	12/7/2019	In progress			
4 United Oil Emergency Generators	Closed Won	\$440,000.00	11/19/2019	Completed			
5 United Oil Standby Generators	Closed Won	\$120,000.00	1/22/2020		✓		
6 United Oil Plant Standby Generators	Needs Analysis	\$675,000.00	12/24/2019				

Account Name United Oil & Gas, Singapore		Account Type Customer - Direct		Industry Energy		Annual Revenue \$1,780,000,000.00	
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private		
1 United Oil Office Portable Generators	Negotiation/Review	\$125,000.00	11/30/2019				
2 United Oil Refinery Generators	Proposal/Price Quote	\$270,000.00	1/14/2020	Yet to begin			
3 United Oil SLA	Closed Won	\$120,000.00	1/21/2020	In progress			

Account Name United Oil & Gas, UK		Account Type Customer - Direct		Industry Energy		Annual Revenue \$240,000,000.00	
Name	Stage	Amount	Close Date	Delivery/Installation Status	Private		
1 United Oil Installations	Closed Won	\$270,000.00	11/25/2019	In progress			

Simply drag the component into Account page and set attributes as below:

- ❖ **Record Id:** empty (gets the page's record Id)
- ❖ **Hierarchy Object Name:** Account
- ❖ **Hierarchy Object Parent Field:** ParentId
- ❖ **Hierarchy Object Main Field:** Name (or another field that identifies an account)
- ❖ **Hierarchy Object Fields:** Type (or as much fields as you need)
- ❖ **Table Object Name:** Opportunity
- ❖ **Table Object Parent Field:** AccountId
- ❖ **Table Object Main Field:** Name (or another field that identifies an opportunity)
- ❖ **Table Object Fields:** Amount (or as much fields as you need)

Save and activate the page. Open an opportunity record page and you should see the combined hierarchy.

USE COMPONENT IN VISUALFORCE

You can use the Combined Hierarchy in a Visualforce page using Lightning Out.

You can find the documentation for using Lightning Components in Visualforce [here](#).

Check this snippet as an example:

```
$Lightning.createComponent("ares:combinedHierarchy",
    // these are the inputs of the component, change values accordingly
    {
        recordId : "identifier of the record to get the hierarchy from",
        treeObjectName: "API name of desired object to be displayed in the tree",
        treeParentFieldName: "API name of the self-lookup field for the hierarchy object",
        treeMainField: "API name of the field used as first column in the tree items",
        treeItemFieldsString: "a comma-separated list of API fields names that you want to see as columns in the tree items",
        tableObjectName: "API name of desired object to be displayed in the tables",
        tableParentFieldName: "API name of the lookup field that link table objects with hierarchy objects",
        tableMainField: "API name of the field used as first column in the tables",
        tableFieldsString: "a comma-separated list of API fields names that you want to see as columns in the table",
        cardIcon: "icon name",
        cardTitle: "the title to be shown in the header of the component",
    },
    // this is the id of the element in the DOM
    "combinedHierarchy",
    // you can remove this callback
    function(cmp) {
        console.log("component was created");
    }
);
```

KNOWN ISSUES

Currently issues that will be addressed in near future:

- Cross object fields are not supported, for example, RecordType.Name or Owner.Name, etc.
Workaround to this problem is to create a formula field and use that formula field in the component.
For example, create a formula field "Record Type Name (Record_Type_Name__c)" and use RecordType.Name in the formula. Use Record_Type_Name__c in the hierarchy columns.

ABOUT ARES

ARES provides Salesforce solutions for customers of all sizes and industries looking for modern and simple apps that solve tough business challenges.

ARES is a project by Alejandro Rocamonde Esteve, a Spain based Salesforce developer. For custom solutions please contact alejandro.rocamonde@proton.me

