



# VisualForce Remote Objects



## EXECUTIVE SUMMARY

Haven't you wondered it would have been better if you could handle all your code for DML operations in one end, eliminating Apex side coding? The concept of "Remote objects" opens the whole new world of opportunity to achieve this. This WP will discuss the benefits of Remote Objects, considerations to keep in mind before you get started and steps to implement this ice breaking feature in the Salesforce world. Keep reading to know more and make your coding job easy and smarter!

## WHITEPAPER

Visualforce Remote Objects are proxy objects that enable basic DML operations on Objects directly from JavaScript. It can be considered as a replacement to JavaScript Remoting as it removes the need for writing @RemoteAction methods in an Apex controller or extension, thereby eliminating some of its complexity.

Implementing this exciting feature will require the below two steps:

- ❖ Write access definitions in VF page with Remote Objects components. This will generate a set of JavaScript proxy objects
- ❖ Define your Data access functions in JavaScript. The proxy objects generated from previous step are used by these functions to perform create, update, retrieve and delete operation on your data

## BENEFITS OF VISUALFORCE REMOTE OBJECTS

- ❖ Every developer must have dreamt of eliminating the process of writing test classes at one point or another in his life. Well, the concept of Remote Objects is a blessing for every developer who have had that thought. You no longer have to write Controllers which means you can easily avoid writing test classes and can do whole of your coding in VF page alone.
- ❖ No more worrying about API limits! VF Pages that use Remote Objects are subject to all the standard Visualforce limits, but like JavaScript remoting, Remote Objects calls don't count toward API request limits.



## DML OPERATIONS USING REMOTE OBJECTS

### 1. CREATING RECORDS

**create()** accepts two arguments, both are optional.

**Syntax:**

```
RemoteObjectModel.create({field_values}, callback_function)
```

**field\_values** - Lets you to define and create a record in one statement

**Callback\_function:**

Lets you to handle the server response asynchronously. It can accept up to 3 arguments.

```
function callback(Error error, Array results, Object event) { // ... }
```

### 2. RETRIEVING RECORDS

**retrieve()** requires two arguments, one for query criteria and one for a callback handler.

**Syntax:**

```
RemoteObjectModel.retrieve({criteria}, callback_function)
```

**field\_values** - Lets you to define and create a record in one statement

**Callback\_function:**

**Criteria** - Remote Objects query object or a function that returns one.

### 3. UPDATING RECORDS

**update()** accepts three arguments, all optional, and can update one or many records at the same time, depending on the arguments that you provide.

**Syntax:**

```
RemoteObjectModel.update([record_ids],  
{field_values}, callback_function)
```

**Record\_ids:**

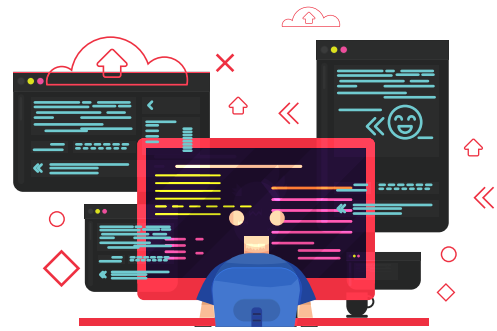
Array of strings, where the strings are the Ids of records to be updated.

### 4. UPSERTING RECORDS

**upsert()** accepts two arguments, both optional.

**Syntax:**

```
RemoteObjectModel.upsert({field_values},  
callback_function)
```



### 5. DELETING RECORDS

**del()** accepts two arguments, both optional, and can delete one or many records, depending on the arguments that you provide.

**Syntax:**

```
RemoteObjectModel.del([record_ids],  
callback_function)
```

**Record\_ids :**

Array of strings, where the strings are the IDs of records to be deleted. If this parameter is not included, the Id that is set on the Remote Object instance will be considered.

## QUERY CRITERIAS FOR REMOTE OBJECT

### 1. WHERE

Following are the operators supported for where condition:

- ◆ **eq-** equals
- ◆ **ne-** not equals
- ◆ **lt-** less than
- ◆ **lte-** less than or equals
- ◆ **gt-** greater than
- ◆ **gte-** greater than or equals
- ◆ **in-** in, used for finding a value that matches any of a set of fixed values. Provide values as an array, for example, ['James', 'Roy', 'Paul']
- ◆ **like-** string matching. As with SOQL, use “%” as a wildcard character
- ◆ **nin-** not in, used for finding a value that matches none of a set of fixed values. Provide values as an array, for example, ['James', 'Roy', 'Paul']
- ◆ **and-** logical AND, used for combining conditions
- ◆ **or-** logical OR, used for combining conditions

## 2. ORDER BY

Just like orderby condition in SOQL, order by conditions in Remote Objects lets you to order the results after retrieve operation.

Maximum limit of fields for ordering is three.

### Syntax:

```
orderby: [ {FieldName: "value"} , {FieldName: "value"} ]
```

### value:

value enables you to sort ascending or descending and to sort null values first or last.

Example: "DESC NULLS FIRST"

## 3. LIMIT AND OFFSET

This condition lets you to retrieve only a specific number of records at a time and to page through an extended set of result.

Default value for limit is 20 and maximum value is 100.

Offset can be used to specify the number of records to skip from overall results before it gets added to the returned list.

Minimum value for offset is 1 and maximum is 2000.

## KEY POINTS TO CONSIDER BEFORE USING REMOTE OBJECTS

- ❖ Remote Objects strictly follows field level security and hence shows blank space for fields that aren't accessible to the person viewing the page. If a user that have no data modification permission tries to perform CRUD operation, process may fail with an error
- ❖ Each CRUD operation is considered as a separate transaction which succeeds and fails on its own. Hence, you have to rethink on using Remote Objects when using it for modifying multiple related objects
- ❖ Consider using server side coding when you have more complex business rules and processes.
- ❖ If testability, performance, security and consistency are your key factors for development, then Remote Objects should not be your choice for the day

## LIMITATIONS

- ❖ You can retrieve a maximum of 100 rows in a single request. To display more rows, submit additional requests by using the OFFSET query parameter.
- ❖ Remote Objects doesn't support Blob fields
- ❖ Setting the rendered attribute to false on Remote Objects components disables the generation of the JavaScript for those Remote Objects

## CONCLUSION

As we saw, Remote Objects opens a whole new approach to the Salesforce development world. Having said that, the choice of whether to use server side coding or Remote objects or JavaScript remoting may depend upon the complexity and nature of your development. However, this Sprint 14 feature is undoubtedly one among the best and lightest approach to perform rapid client side querying without impacting API limits.

## SOURCE

 [https://developer.salesforce.com/docs/atlas.en-us.pages.meta/pages/pages\\_remote\\_objects.htm](https://developer.salesforce.com/docs/atlas.en-us.pages.meta/pages/pages_remote_objects.htm)



Suyati is a fast-growing, digital transformation solutions company that helps you rebuild your customer experience for the digital consumer. We collaborate with businesses to strategize and implement impactful digital initiatives that position our clients ahead of the competition. We are digital-first and we focus on delivering digital transformation solutions that support your various engagement strategies.

Our three-phase approach to implementing digital transformation for you ensures that you win stakeholder support, secure early wins through competitive advantage, and transform your business for future growth. And our tailor-made platform, Mekanate, helps you discover your business DNA from your passive and active data, and use it to initiate, integrate and accelerate your DT implementation.

With our niche and rich expertise in a wide range of technologies and services - CMS, CRM, e-commerce, Cloud, IoT, Data Analytics, and Product Engineering - we help companies across the globe leverage their best on web/cloud/mobile platforms.

Learn more: [www.suyati.com](http://www.suyati.com)

Get in touch: [services@suyati.com](mailto:services@suyati.com)

