5.21.4 Loading Bookmarks

You can load a bookmark at runtime, also called “to apply a bookmark”, instantly via scripting and keep the existing widget states from the users’ previous interactions for any not-included widgets (for example, selections, filters, drilldowns, sortings, rankings, expanding and collapsing states of charts and tables, input values of input fields or sliders).

When applying a bookmark, the operation is applied to the bookmark's *Included Components* configured at design time.

```
// Apply the bookmark to the current analytic application. Returns false if the
bookmark isn't available to the current user. The input is the bookmark ID or a
BookmarkInfo object.
BookmarkSet_1.apply(bookmark);
```

When applying a bookmark in an analytics designer script at runtime, then the application is reloaded.

5.21.4.1 Tip: Loading Bookmarks in the onInitialization Event Script

When you load a bookmark at runtime with the `apply()` script API method in the `onInitialization` event script, then the analytic application is reloaded and the `onInitialization` event script is executed again. This can lead to an infinite loop. The following example shows one way to avoid this:

Example:

In the following snippet from an `onInitialization` event script, the first bookmark is applied to the analytic application without causing an infinite loop:

```
var bookmarks = BookmarkSet_1.getAll();
if ((BookmarkSet_1.getAppliedBookmark() === undefined) && (bookmarks.length > 0)) {
  var bookmark = bookmarks[0];
  BookmarkSet_1.apply(bookmark);
}
```

5.21.5 Getting Bookmark Information

You can retrieve information about bookmarks with the `getAll()`, `getVersion()`, and `getAppliedBookmark()` script API methods. You can check if the state of the current application is different from its state captured in a bookmark with the `isSameAsApplicationState()` script API method.

Example:

In the following example, see some applications of the described script API methods:

```
BookmarkSet_1.getAll();      // get all valid bookmarks
BookmarkSet_1.getVersion();  // get the version of current bookmark

// get current applied bookmark
var bookmarkInfo = BookmarkSet_1.getAppliedBookmark();

// check if bookmark is changed
BookmarkSet_1.isSameAsApplicationState(bookmarkInfo);
```