



## Advanced Themes

**Initial Release Date:** 21st April 2008

**Document Modification Date:** 21st April 2008



## Introduction

Themes allow you to apply formatting to the Spark look and feel. However, they can also be used to provide more dynamic feedback to the end user – such as bread-crumbs or shared Spark values. In this document, we outline a few techniques that are helpful to make your themes more dynamic and engaging for the end user.

## Reference

Firefly Web Server exposes a number of values that are useful for Theme designers as standard. These values are exposed as hidden form input values:

```
<input type="hidden" name="name" id="id" value="value" />
```

The standard values are:

- **spark\_name**: The name of the Spark currently being viewed by the user.
- **firefly\_item\_name**: The name of the Element currently being viewed by the user.

If the Spark contains Groups, the following are also included on the page:

- **group\_name**: The name of the Group the user is currently navigating
- **group\_guid**: The GUID for the Group the user is currently navigating

Firefly Web Server can also expose Variable Element values within the page. These values are also exposed as hidden form input values:

```
<input type="hidden" name="expose_[variablename]" id="expose_[variablename]" value="The value of the variable" />
```

E.g. for a Variable called “CustomerTelephone”, this would be exposed as:

```
<input type="hidden" name="expose_customertelephone" id=" expose_customertelephone" value="123-456-7890" />
```

This allows Theme designers to use these values as part of the Theme content. The following rules are applied to the Variable Name:

1. It has “expose\_” put in front of the name as it is defined in the Spark
2. It puts the name of the variable into lower case – entirely
3. If the Variable name contains any spaces, these are replaced with a “\_” character

Therefore a variable called “Customer Name” would be re-named “expose\_customer\_name”.

## Showing Variable Element values within the Theme

Sometimes it is useful to show values contained in the Spark on every screen. You can already do this using Resource Elements within the Spark – however, this can often become a laborious task of including the same Resource reference at the top of every visual Element (i.e. Question, Statement, etc).



To illustrate the usefulness of showing values within a Theme, we'll use an example. Imagine you are running a customer contact centre and you wish to display the customer contact details at the top of every screen. These details are stored in Variable Elements within the Author's Spark. Rather than having the Author reference these variables within the Spark and manually creating the contact details section on every screen, it would be more helpful to put this information within the Theme. Here's how you do it.

Within Designer,

1. The customer contact details in the Spark must be stored in Variable Elements – e.g. CustomerName, CustomerTelephone etc. If this is not the case, you'll need to create Variables and put the values from your Question, into these Variables using the Allocator. Please refer to the Designer documentation for help on this.
2. When defining your Variable Element, click on the 'Settings' tab and tick the box for 'Expose this variable to theme designers (Web Server only). This will make the Variable value available to the Theme when the Spark is run on the Web Server.
3. Publish the Spark to the Web Server with your changes.

Within the Web Server,

1. You need to edit the Theme for the Space which contains the Spark.
2. Under the Main tab, add the following JavaScript to the header:

```
<script type="text/javascript" language="javascript">

function writeHeaderInfo() {

    var sparkName = 'Spark Name: ' + document.getElementById('spark_name').value;
    var itemName = 'Item Name: ' + document.getElementById('firefly_item_name').value;
    var customerName = 'Customer Name: '
        + document.getElementById('expose_customername').value;
    var customerTelephone = 'Customer Telephone: '
        + document.getElementById('expose_customertelephone').value;

    document.getElementById('display_spark_name').innerHTML = sparkName;
    document.getElementById('display_firefly_item_name').innerHTML = itemName;
    document.getElementById('display_customer_name').innerHTML = customerName;
    document.getElementById('display_customer_telephone').innerHTML = customerTelephone;
}

</script>
```

3. Add the following onload command to the body tag:

```
onload="javascript: writeHeaderInfo();"
```

4. Within the content of your Spark, add the following div's (this is where the content will be displayed):

```
<div id="display_firefly_item_name"></div>
<div id="display_spark_name"></div>
<div id="display_customer_name"></div>
<div id="display_customer_telephone"></div>
```

5. Save the Theme and run the Spark.