



LUMIRA 2.1 KOMMENTIERUNG

Technische Komponente

INHALT

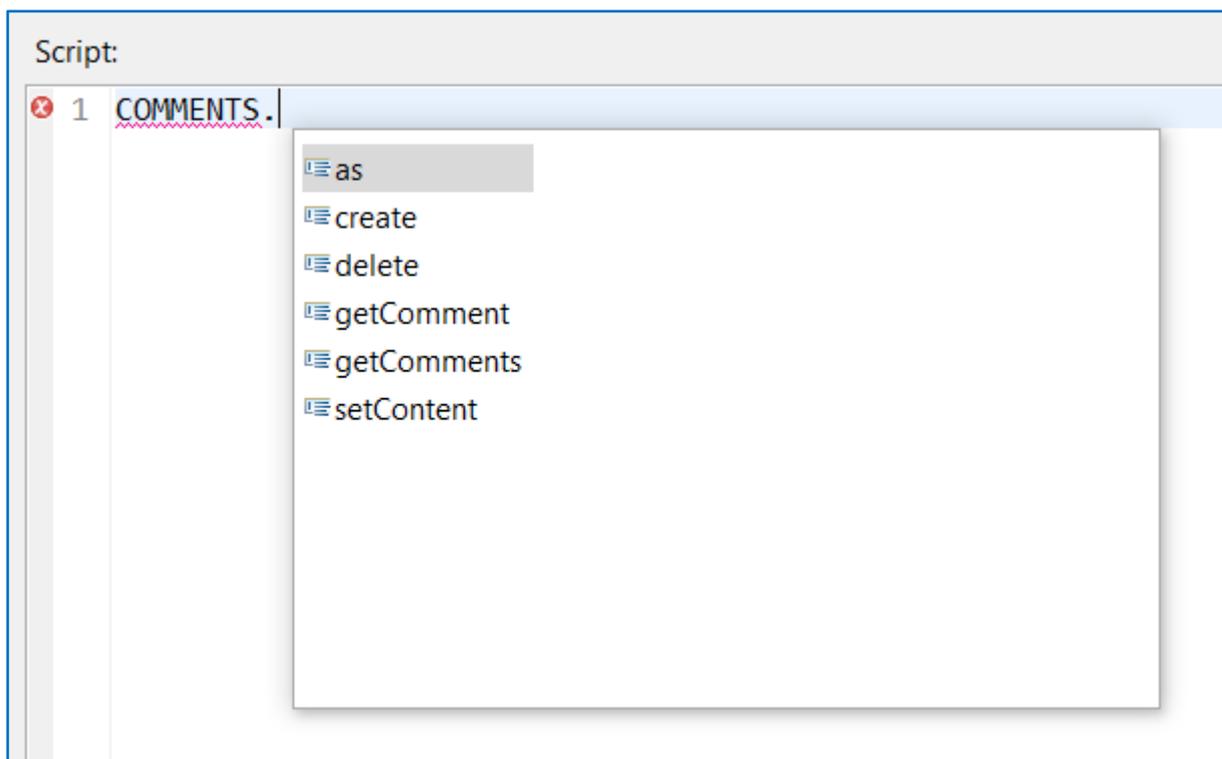
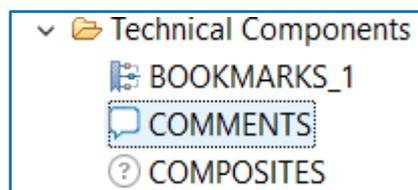
1	Einführung	3
2	Kommentar erstellen	4
3	Kommentar löschen.....	6
4	Kommentar auslesen	6
5	Kommentar updaten	7
6	Kommentarablage	7
7	Beispiel zum Speichern von Kommentaren in einer Crosstab	9

1 Einführung

Kommentierungen von Berichten sind heutzutage nicht mehr weg zu denken. Um agil auf Veränderungen reagieren zu können, ist Kollaboration und schneller Informationsaustausch wichtig. Oft ist dies aber im Standard nicht möglich und komplexe Anwendungen müssen aufwändig installiert und gepflegt werden.

Die, mit Release von Lumira 2.1 veröffentlichte, technische Komponente für Kommentierung erlaubt es, Kommentare direkt im Dashboard zu erfassen und zu teilen. Dieses lang ersehnte Feature wird in diesem Beitrag technisch genau erklärt und zeigt praxisbezogen wie die neuen Kommentierungsmöglichkeiten eingesetzt werden können.

Um die neue Funktionalität einzusetzen, muss die technische Komponente in die Applikation aufgenommen werden. Danach stehen alle relevanten Scripting Methoden zur Verfügung.



2 Kommentar erstellen

Beim Erstellen eines Kommentars kann festgelegt werden, auf welchen Kontext dieser gespeichert werden soll, um ihn später wieder abzurufen.

```
COMMENTS.create(content, {  
    key1: value1,  
    key2: value2  
});
```

Der Inhalt / Text [content] des Kommentars wird zu einem vordefinierten Kontext gespeichert.

create (content, values?)

Creates a comment.

Parameters

Name	Type	Description
content	String	Content of the new comment
(optional) values	CommentCreationValues	Details of the new comment

Returned Value

String. It contains the comment ID of the newly created comment.

Die "CommentCreationValues" enthalten folgende Eigenschaften, um den Kommentar einzigartig zuzuweisen:

CommentCreationValues

An object specifying comment properties when creating a comment.

Fields

Name	Type	Description
context	ResultSetSelectionByString	ResultSetSelectionByString. Context that can be used to associate the comment with a specific result set selection
contextType	CommentContextType	CommentContextType. Type of the context
dataSource	DataSourceAlias	DataSourceAlias. Data source that can be used as a query parameter for retrieving comments
isPublic	Boolean	Boolean. Specifies whether this comment is public, and thus available to users other than the author

context: Das Feld gibt an, auch welche Kombination der Kommentar gespeichert wird. Einige Beispiele wären entweder Applikationsübergreifend, auf eine vorherrschende Filtereinstellung oder eine gewählte Datenzeile innerhalb der Crosstab.

contextType: Zur weiteren Definition kann hier der Kommentartyp angegeben werden. Es sind folgende Typen vorgesehen:

CommentContextType

A set of constants to specify the context type of the comment.

Constants

Name	Description
CommentContextType.DATA	Comment applies to a data cell
CommentContextType.CONTEXT	Context of a comment is bound to applied filters of a specified list of dimensions
CommentContextType.MEMBER	Comment applies to a member
CommentContextType.DIMENSION	Comment applies to a dimension
CommentContextType.NONE	Comment has no context. This value is useful if you want to get all the comments without a specific context type.

CommentContextType.DATA: Kommentaruweisung zu einer spezifischen Datenzelle innerhalb einer Crosstab.

CommentContextType.CONTEXT: Zuweisung zu einem vordefinierten Kontext, als Beispiel eine Filtereinstellung der zu kommentierenden Crosstab oder der Applikationsname.

CommentContextType.MEMBER: Zuweisung zu einem Member der Crosstab.

CommentContextType.DIMENSION: Zuweisung zu einer Dimension der Crosstab.

CommentContextType.NONE: Wenn kein Kontext zu dem Kommentar zugewiesen werden soll.

dataSource: Speichert den Kommentar zu einer spezifischen DataSource innerhalb der Applikation.

isPublic: Ein boolean der spezifiziert, ob der Kommentar für alle zugänglich ist (public) oder nur von der erstellenden Person (private). Diese Einstellung hat nichts mit Berechtigungen zu tun, sondern ist eine reine Lumira spezifische Definition.

Hier ein Beispiel, wie ein Kommentar für die Applikation erstellt werden kann. Die Zuweisung erfolgt über den Applikationsnamen und ist für alle Anwender lesbar.

```
var context = "{Application:" + APPLICATION.getInfo().name + "}";  
COMMENTS.create("Beispielkomment für Applikation",{  
    "context": context,  
    "contextType": CommentContextType.NONE,  
    "dataSource": DS_1,  
    "isPublic": true  
});
```

3 Kommentar löschen

Jeder gespeicherte Kommentar hat eine eindeutige ID, mit welcher der Kommentar auch wieder gelöscht werden kann.

delete(commentId)

Deletes a comment.

The comment ID must have been created by the current comment component.

Parameters

Name	Type	Description
commentId	String	Comment ID

Returned Value

None

4 Kommentar auslesen

Das Auslesen eines einzelnen Kommentars geschieht ebenfalls über die ID.

getComment(commentId)

Returns a comment.

The comment ID must have been created by the current comment component.

Parameters

Name	Type	Description
commentId	String	Comment ID

Returned Value

Comment. It contains a comment.

Eine weitere Möglichkeit besteht darin, mehrere Kommentare auszulesen anhand des Kontextes der definierten Kommentare.

getComments(values?)

Returns comments. You can optionally specify comment query values to narrow down the set of the returned comments.

Parameters

Name	Type	Description
(optional) values	CommentQueryValues	Comment query values to narrow down the set of the returned comments

Returned Value

Array of Comment. It contains comments.

Hier ein Beispiel, wie die Kommentare, welche für die Applikation erstellt wurden, ausgelesen werden können. So erhalten wir alle Kommentare, welche auf dem spezifizierten Kontext abgespeichert wurden im array ausgegeben.

```
COMMENTS.getComments({
    "context": context,
    "contextType": CommentContextType.CONTEXT,
    "dataSource": DS_1,
    "isPublic": true,
    "order": SortOrder.ASCENDING,
    "sortBy": CommentSortBy.CREATED_ON
});
```

5 Kommentar updaten

Kommentare können auch nachträglich geändert werden. Dies geschieht über die folgende Funktion.

setContent(commentId, content)
Sets the content of a comment.

Parameters

Name	Type	Description
commentId	String	Comment ID
content	String	Comment content

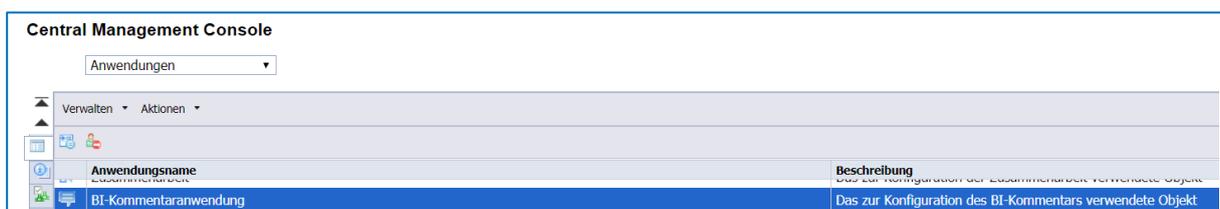
Returned Value
None

6 Kommentarablage

Die Kommentierung kann sowohl lokal, als auch auf der BI Plattform verwendet werden. Lokale Kommentare werden im Lumira repository gespeichert und können dort weiter analysiert werden.

C:\Users\[USER]\LumiraDesigner-workspace\com.sap.ip.bi.zen\repository\
__TECHNICAL_CONTENT\COMMENT

Kommentare in Applikationen auf der BI Plattform werden direkt in der in der BI-Kommentaranwendung definierten Datenbank gespeichert. Kommentare können ab 4.2 gespeichert werden. Die Ablage kann in der CMC überprüft werden.



Die Einstellung hier speichert die Kommentare in der Audit-Datenbank. (hier: MS SQL)

Eigenschaften: Kommentar

Audit-Datenbank verwenden Andere unterstützte Datenbank verwenden

Konfiguration

Kommentar-Datenbank konfigurieren

Typ

Datenbankname

Host

Port

Benutzername

Kennwort

Alle Kommentare löschen, die älter sind als Tage

Nachdem Sie die Änderungen in der Kommentar-Datenbank gespeichert haben, müssen Sie den APS-Server neu starten.

Beispiele von gespeicherten Kommentaren.

1	A	B	C	E	M	
CommentID	DocId	CommentText	UserName	AppContext		AppDatapath
31	83 AqNqG4yM7bQfFyY50Lx0Vc	New Comment	T80*100/ENZM			
32	84 AqNqG4yM7bQfFyY50Lx0Vc	Dies ist ein neuer Kommentar. Was sind die Eigenschaften?	T80*100/ENZM			
33	85 AqNqG4yM7bQfFyY50Lx0Vc	Diesmal auf Filtereinstellungen von Schland und QZ.	T80*100/ENZM	:NONE;DS_7;003N854375G61835C19U610L;DCALQUARTER;DSRHOTEL_DSRCOLTRY		I003N854375G61835C19U6401Z,I003N854375G61835C19U64L

7 Beispiel zum Speichern von Kommentaren in einer Crosstab

DSRHOTEL Calendar Year/Month	Auslastung (%)		
	APR 2017	MAY 2017	JUN 2017
Sunny Rooms Berlin	74.1	74.1	77.4
Sunny Rooms Düsseldorf	84.8		
Sunny Rooms Frankfurt	76.6	76.6	80.1
Sunny Rooms Friedrich	68.3	68.3	71.3
Sunny Rooms Hamburg	86.6	86.6	90.0
Sunny Rooms München	74.4	74.4	77.6
Sunny Rooms Potsdam	75.4	75.4	78.6
Sunny Rooms Usedom	80.2	80.2	84.4
Sunny Rooms Weimar	72.4	72.4	75.7

Berlin in April mässige Auslastung auf Grund des schlechten Wetters.

1. Kommentar auf Datenzelle erfasst. Durch die neu hinzugekommene Anzeigeeinstellung der Crosstab, können die Kommentare direkt innerhalb der Crosstab angesehen werden. Per Mouseover kann der eingegebene Kommentar, indiziert durch das rote Dreieck oben rechts, gelesen werden.

Conditional Form	
Comments Visib	true
Title Display	None
Title Text	Crosstab Title

2. Kommentar auf Member erfasst.

3. Kommentar auf Dimension erfasst.

Für die Kommentierung auf Tabellenebene bietet es sich an, mit Hilfe vom Kontextmenu auf die ausgewählten Bereiche eingeschränkt. Dazu wurde die technische Komponente CONTEXT_MENU hinzugefügt und ein neuer Eintrag für Kommentare erstellt.

DSRHOTEL Calendar Year/Month	Auslastung (%)		
	APR 2017	MAY 2017	JUN 2017
Sunny Rooms Berlin	74.1	74.1	77.4
Sunny Rooms Düsseldorf	84.8	84.8	89.3
Sunny Rooms Frankfurt	76.6	76.6	80.1
Sunny Rooms Friedrich	68.3	68.3	71.3
Sunny Rooms Hamburg			90.0
Sunny Rooms München			77.6
Sunny Rooms Potsdam			78.6
Sunny Rooms Usedom			84.4
Sunny Rooms Weimar			75.7

- Filter
- Filter and Remove Dimension
- Exclude
- Drill Path...
- Conditional Formatting
- Kommentar

```

var selection = CONTEXT_MENU.getSelection();
var selectionType = CONTEXT_MENU.getSelectionType();

APPLICATION.createInfoMessage("Selectiontype: " + selectionType);

if (selectionType == "DATA") {
    selection = DS_7.getBackgroundFilters(selection);
}

selection.forEach(function(element, key) {
    element.forEach(function(element_2, index) {
        APPLICATION.createInfoMessage("Selection: " + element_2 );
    });
});

var text = TEXTAREA_1.getValue();

COMMENTS.create(text, {
    "context": selection,
    "contextType": selectionType,
    "dataSource": DS_7,
    "isPublic": true
});

```

Das if-statement wird benötigt, wenn auf Datenzelebene eingeschränkt werden soll, da ev. vorhandene Filter benötigt werden, um den Kommentar in der Crosstab ersichtlich zu machen. Der eingegebene Text in "TEXTAREA_1" wird dann als Kommentar abgespeichert.

Haben wir Ihr Interesse geweckt?

Gerne besprechen wir mit Ihnen Ihre individuellen Anforderungen.
Ihr Ansprechpartner für:

Schweiz

Roland J. Merz
Managing Director
Tel: +41 79 285 71 00
roland.merz@cubeserv.com

Deutschland

René Lechtenböhrer
Mitglied der Geschäftsleitung
Tel.: +49 173 2146155
rene.lechtenboehmer@cubeserv.com

Österreich

Mag. Marcel Suppan-Luttenberger
Head of Division Austria
Tel.: +43 664 8825 1793
marcel.suppan@cubeserv.com

Kroatien

Dražen Pavlović
Head of Region /
Member of the Management Board
Tel.: +38 591 3871201
drazen.pavlovic@cubeserv.com

Version Oktober 2019