CBM development guide

# Agreements:

## Naming conventions:

* 1. Link relations might be named as pointed *Concept*(/or *view*), or the same prefixed with “For”<concept\_name> (the last usually for back-links). If there are several links to the same *Concept*, this rule is obviously not applicable, but in other cases it is recommended to follow.

# Code Snippets:

“Create from” snippet

Used for wide range of cases, from pure many-to-many relation, thru reified n-ary association and to pure one-to-many associated instances only initiated based on info from third concept entities.

Is placed to

[{

title: '###TITLE###',

showHover: true,

cellHover: '###TIP\_TEXT###',

icon: isc.Page.getAppImgDir() + 'add.png',

click: function (topElement) {

var tbl = createTable('###CONCEPT\_FROM\_WHICH\_CHOOSE###',

arguments[0].context,

this.###FUNC\_TO\_INIT\_CREATION###,

null,

arguments[0].context.topElement.valuesManager.getValue('ID'),

null

);

},

###FUNC\_TO\_INIT\_CREATION###: function (srcRecords, context) {

createFrom(srcRecords, function (srcRecord) {return 'CategoryOffer';},

###CREATED\_CONCEPT\_NAME###.CreateFromMethods[0].###FUNC\_TO\_CREATE\_SINGLE\_INSTANCE###,

context);

},

###FUNC\_TO\_CREATE\_SINGLE\_INSTANCE###: function (dstRec, srcRec, mainID) {

// For many-to-many:

dstRec.###FIELD\_POINT\_TO\_\_CONCEPT\_FROM\_WHICH\_CHOOSE### = srcRec.ID;

}

}]

And example:

[{

title: 'Выбрать контакты',

showHover: true,

cellHover: 'Выбрать контакты из списка',

icon: isc.Page.getAppImgDir() + 'add.png',

click: function (topElement) {

var organization = arguments[0].context.topElement.valuesManager.getValue('Organization');

var tbl = createTable('Contact',

arguments[0].context,

this.createBoundContacts,

{Organization: organization},

arguments[0].context.topElement.valuesManager.getValue('ID'),

null

);

},

createBoundContacts: function (srcRecords, context) {

createFrom(srcRecords, function (srcRecord) {return 'BoundContact';},

BoundContact.CreateFromMethods[0].createBoundContactFromContact,

context);

},

createBoundContactFromContact: function (dstRec, srcRec, mainID) {

dstRec.Contact = srcRec.ID;

var org = isc.DataSource.get('Organization').getCacheData().find({ID: srcRec.Organization});

dstRec.Organization = org.Description;

}

}]