Sizmek™

Convert HTML5 Adobe Edge Animate project to Sizmek Compatible Ad

Overview

This conversion guide is a step-by-step outline of the processes required to quickly and easily convert an existing adobe edge project into a Sizmek compatible ad. It is not a tutorial on how to use Adobe edge to build HTML5 ads, and is intended for use once you have completed building your Adobe Edge Animate project – and are ready to convert it for Sizmek ad serving.

This guide is for Standard/Polite banners only – and does not cover Expandable banners.
If you are building a HTML5 expandable banner please contact our creative team at creativeau@sizmek.com

For a deeper understanding of some of the code referenced in this guide, please also read our build guides for Sizmek HTML5 Ads available by contacting creativeau@sizmek.com

Brief overview of steps/checklist

STEP 1: Add EdgePreview_EBLoader.js to project library.

STEP 2: Prevent stage from auto-playing

STEP 3: Add Sizmek initialization code in ‘document.compositionReady’

STEP 4: Add Clickthrough and/or Custom Interactions on elements/events.

STEP 5: Preview and Debug.

STEP 6: Prepare/package contents for upload.

a) Edit the Index.html file and add our external scripts.
b) Prepare Project Folder
c) Upload to platform

APPENDIX: Common Issues Troubleshooting

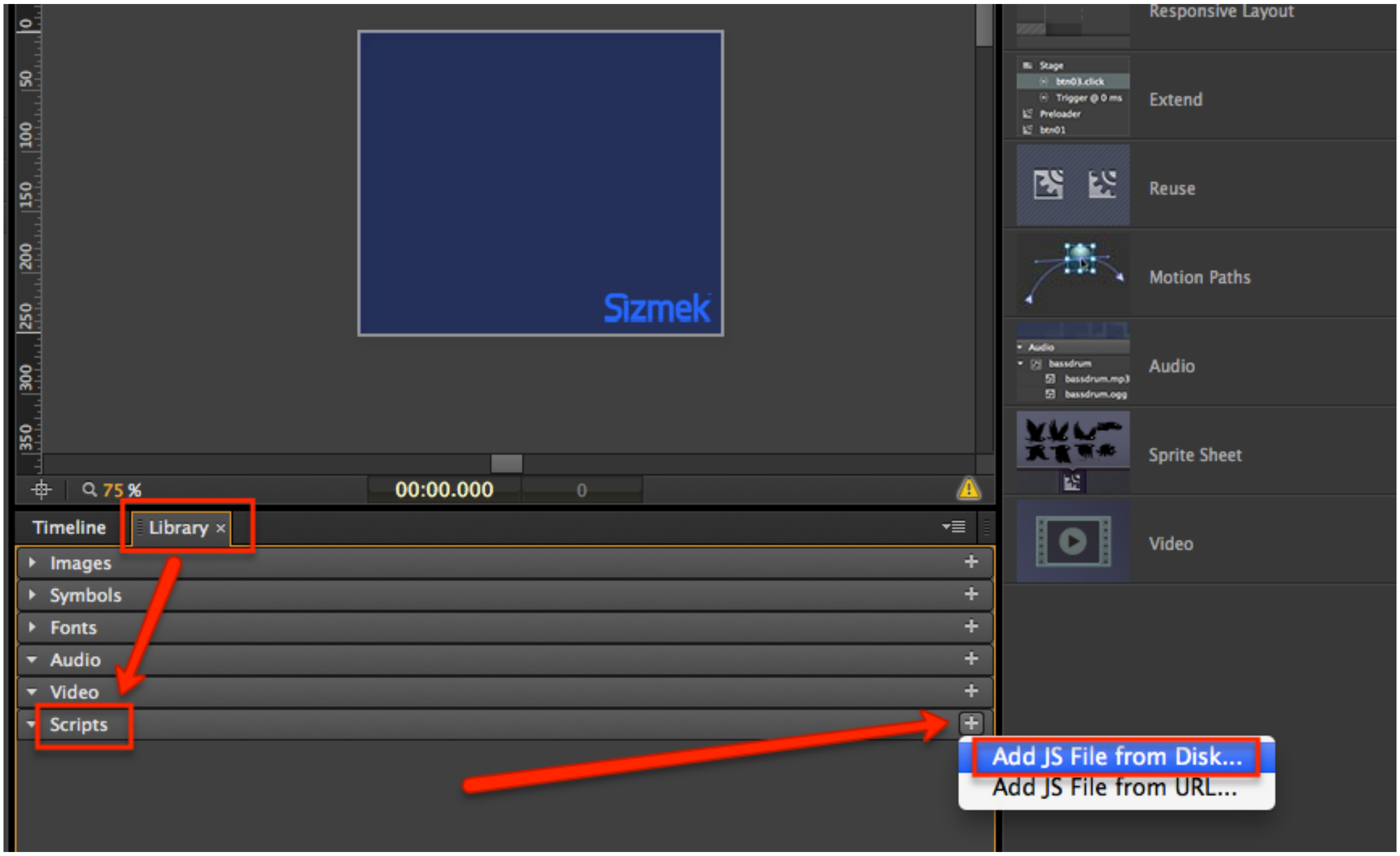
Detailed Overview:

STEP 1: ADD EdgePreview_EBLoader.js FILE TO THE PROJECT LIBRARY

In order for the sizmek code to work **locally** you will need to manually add our EdgePreview_EBLoader.js file to the project.

This script allows you to test and preview the ad locally and run Sizmek code for preview.

In Adobe Edge, open the Library. (Window > Library)
In the scripts section, add a script (+) > ‘add JS file from Disk.



Link to the ‘EdgePreview_EBLoader.js’ file in the project template folder.
If you do not have this script, you can download it here:
http://demo.mediamind.com/australia/customjs/EdgeConversionScripts.zip

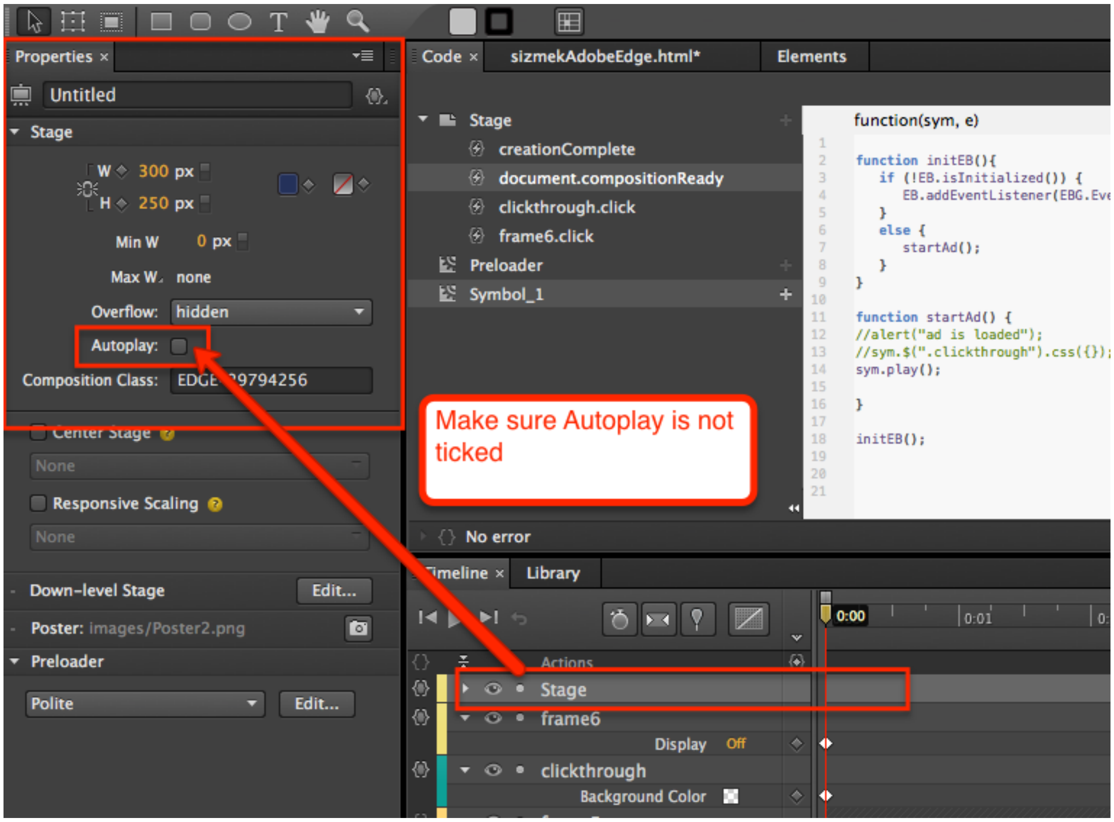
STEP 2: Prevent the stage from auto-playing.

In step 3 we will add code which checks for the Sizmek classes to load before starting the stage animation.
In order for this to work properly, we need to prevent the stage from auto playing (default setting)

Make sure the Stage is selected in the Timeline.

Open the properties tab.

Un-tick the Autoplay box.



Make sure Autoplay is not ticked

Properties

Untitled

Stage

W 300 px

H 250 px

Min W 0 px

Max W none

Overflow: hidden

Autoplay: □

Composition Class: EDGE-29794256

□ Center Stage

None

□ Responsive Scaling

None

Down-level Stage Edit...

Poster: images/Poster2.png

Preloader

Polite Edit...

Code × sizmekAdobeEdge.html* Elements

- Stage
  - creationComplete
  - document.compositionReady
  - clickthrough.click
  - frame6.click
- Preloader
- Symbol_1

function(sym, e)

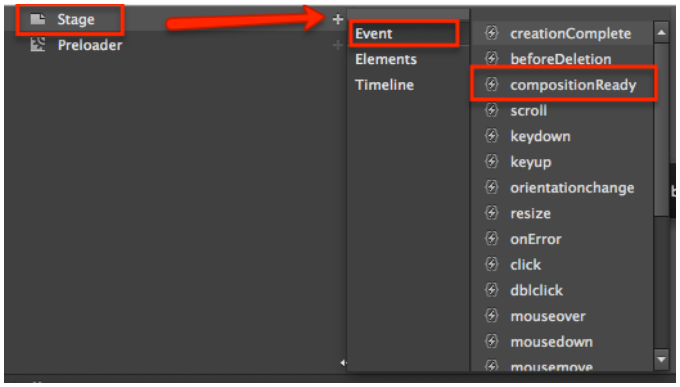
```
function initEB(){
   if (!EB.isInitialized()) {
      EB.addEventListener(EBG.Eve
   }
   else {
      startAd();
   }
}

function startAd() {
//alert("ad is loaded");
//sym.$(".clickthrough").css({});
sym.play();

}

initEB();
```

No error

STEP 3: Add our initialization code in 'document.compositionReady'

If you don't already have it open, open the event document.compositionReady:

Open Code panel (Window > Code).
Select: Stage and then + .
Select: Event > CompositionReady



Within the compositionReady panel, copy and paste the following code:

```
// Sizmek function to check if Sizmek classes are initialized.
//if initialized, call startAd else add listener

function initEB(){
        if (!EB.isInitialized()) {
                EB.addEventListener(EBG.EventName.EB_INITIALIZED, startAd);
        }
        else {
                startAd();
        }
}
//called once Sizmek classes are initialized
function startAd() {

//any functions to run on load should go here

//play the stage/animation
sym.play();
}

//check if Sizmek is initialized
initEB();
```

This set of code checks to see if EB is initialized (if it is connected to the Sizmek) and if so, calls the function to startAd.

Within this function you can place any code that is meant to run on page load.

sym.play() is called to start the stage animation now that the Sizmek classes are loaded.

STEP 4: Add Clickthrough and/or Custom Interactions on elements/events.

The functions for custom interactions are below:
(These ARE case sensitive).

Clickthrough	EB.clickthrough();
User Action Counter	EB.userActionCounter(“nameOfUserActionCounter”);
Automatic Event Counter	EB.automaticEventCounter(“nameOfUserActionCounter”);

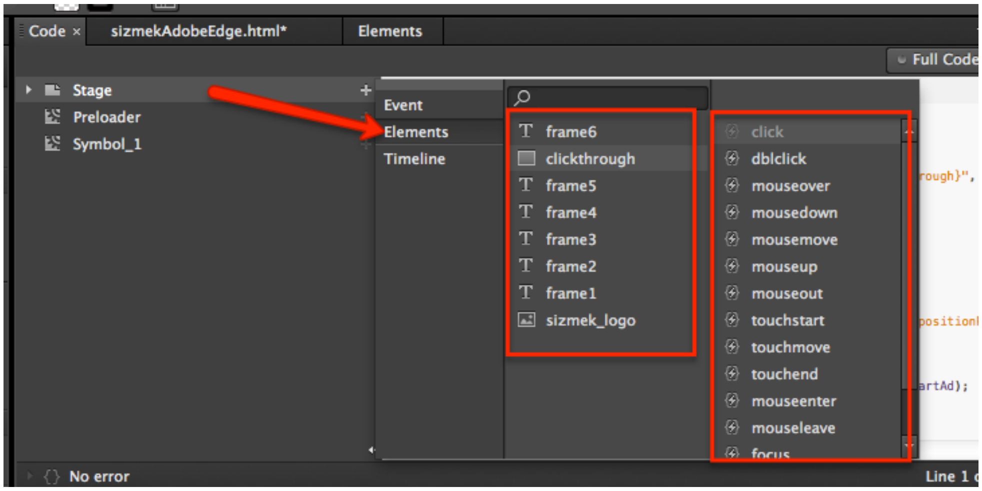
For more information on HTML5 functions: https://support.sizmek.com/hc/en-us/articles/201128145-Sizmek-HTML5-API

Add these functions to perform a clickthrough, or to track certain interactions/events within the events created through edge.

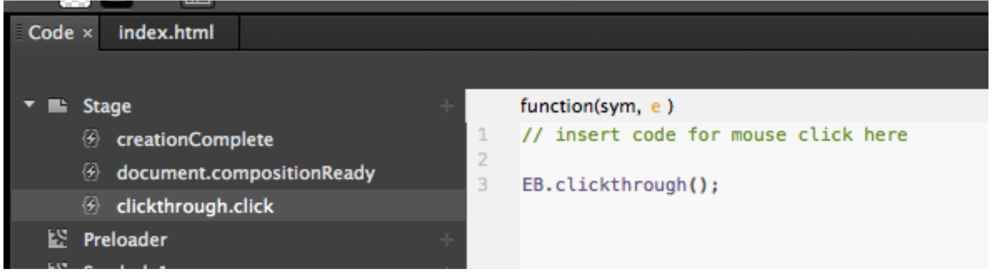
EXAMPLE:

Open Code Window (Window > Code)
Select Element and add a click function:

(Code > Stage + > Elements > NameOfElement > click)



Within this click function, paste EB.clickthrough();



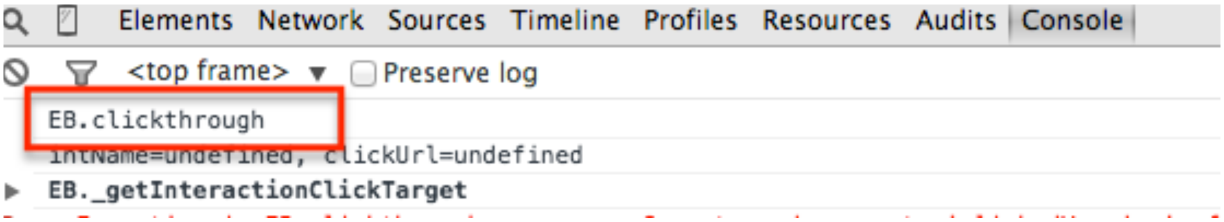
```
function(sym, e )
// insert code for mouse click here

EB.clickthrough();
```

STEP 5: Preview and Debug

Within adobe edge, select to preview the project (in a browser).

In the browser, open a developer console – or install the Adobe Edge Inspector browser plugin here: https://creative.adobe.com/products/inspect

Check the console output to determine that the ad is successfully connecting, and clickthroughs/custom interactions are being tracked.



```
EB.clickthrough
intName=undefined, clickUrl=undefined
▶ EB._getInteractionClickTarget
```

(PLEASE NOTE: Because the ads are not yet hosted on Sizmek, the console will log errors for Clicked Version of Null. Once uploaded to the Sizmek System, these errors will no longer occur).

STEP 6: Prepare/Package contents for upload.

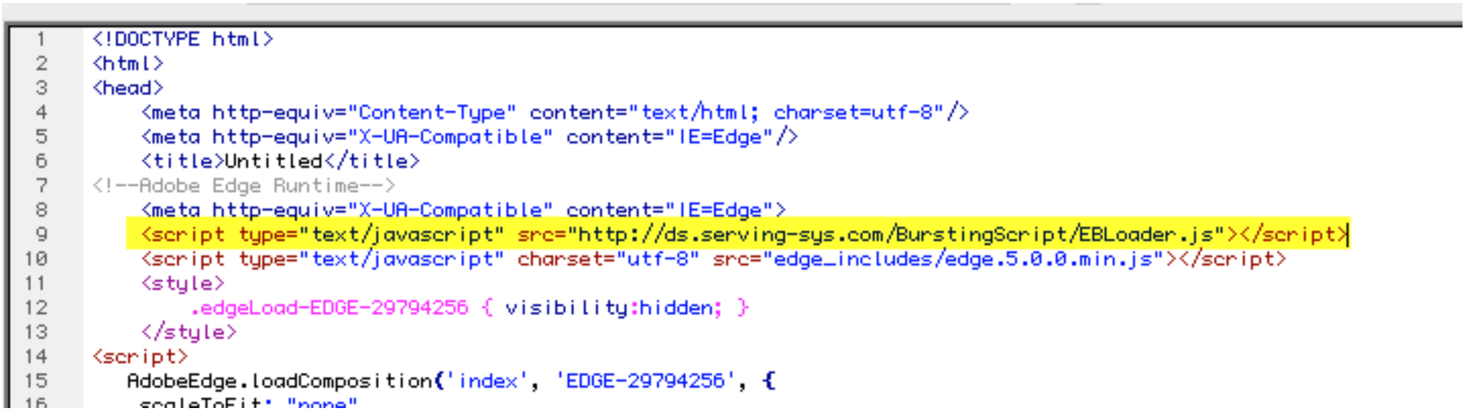
A) Edit the html file and add our external EBLoader.js script

In your project folder, adobe edge will automatically generate a HTML file.
Open this file in an html editor (notepad, dreamweaver etc).

For the ad to work in our platform you will need to add the Sizmek EBLoader Javascript file so that the ad can connect to our classes.

Copy and paste the following code into the <head> section of the html file.

<script type="text/javascript" src="http://ds.serving-sys.com/BurstingScript/EBLoader.js"></script>

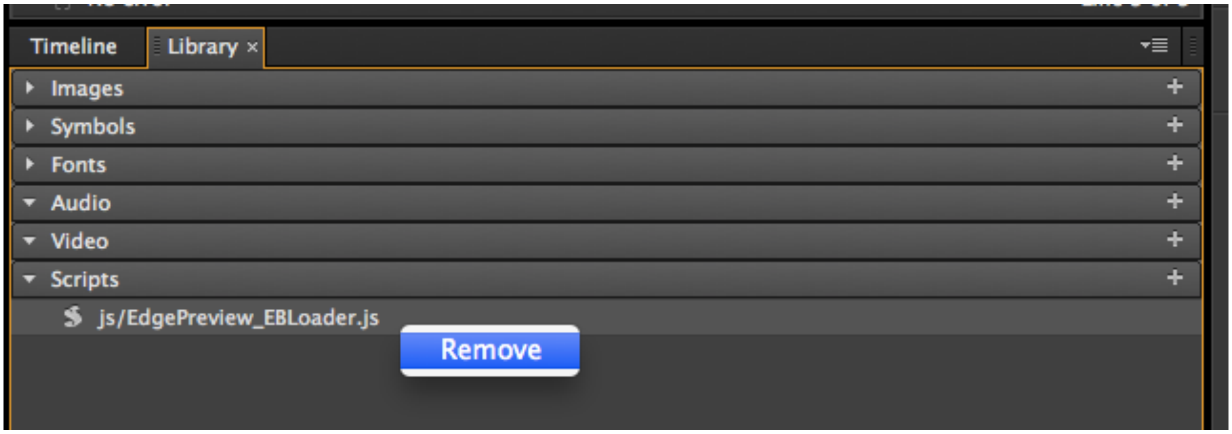


```
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <meta http-equiv="X-UA-Compatible" content="IE=Edge"/>
    <title>Untitled</title>
<!--Adobe Edge Runtime-->
    <meta http-equiv="X-UA-Compatible" content="IE=Edge">
    <script type="text/javascript" src="http://ds.serving-sys.com/BurstingScript/EBLoader.js"></script>
    <script type="text/javascript" charset="utf-8" src="edge_includes/edge.5.0.0.min.js"></script>
    <style>
        .edgeLoad-EDGE-29794256 { visibility:hidden; }
    </style>
<script>
   AdobeEdge.loadComposition('index', 'EDGE-29794256', {
    scaleToFit: "none"
```

B) Remove the EdgePreview_EBLoader.js Script from Adobe Library

1) The EdgePreview_EBLoader.js file has to be removed from the project before it is uploaded to the platform so that it doesn't conflict with the EBLoader.js file added in step A.

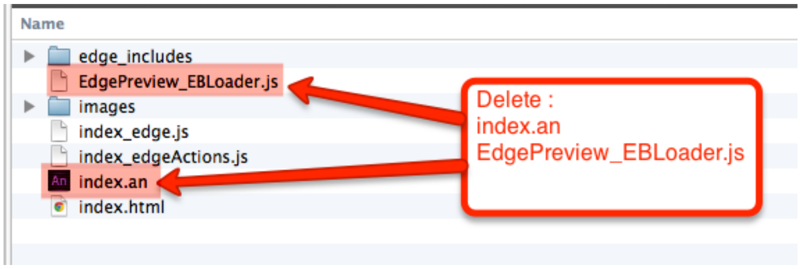
Select the script in the library.
Right-Click > Remove



B) Prepare project folder

1) Make sure that your file is named/saved as index (this needs to be done within adobe edge (Save As), otherwise the logic of the code will not work if it is referencing a differently named html file). This will correctly output the html project file as index.html (which is a requirement for all Sizmek HTML5 ads).

2) Remove the index.an file from the root folder (but keep the **index_edge.js** & **index_edgeActions.js** & the **edge_includes** folder).



3**) If you are building a rich/polite ad** – make sure to copy the scripts folder (which contains the rich_banner.js file) that came with the template. Or download here: http://demo.mediamind.com/australia/customjs/EdgeConversionScripts.zip
You will also need to link to the rich_banner.js file underneath the EBLoader.js file in the html as shown in step A (make sure it is linked BELOW the EBLoader.js <script>.

4) Make sure you have **a back up image** in the folder. This back up image must be the EXACT dimension of your banner.

C) Upload to Platform

1) Zip the root folder and upload to the platform as a creative asset. This will create the ‘workspace’.
2) Create New ad. (Shortcuts > Create New Ad)
3) Select Ad format as either ‘HTML5 Standard Banner’ or ‘HTML5 Polite Banner’ (depending on your specs).
4) Link to the workspace by selecting the zip file in your creative assets list.
5) Select back up image from within the workspace folder.
6) Save
7) Preview ad and check that ad is loading and interactions are tracked.

For detailed information on uploading HTML5 ads please refer to our HELP section
https://support.sizmek.com/hc/en-us/articles/200722459
or contact creativeau@sizmek.com

APPENDIX: Common Issues Troubleshooting

Because the stage is no longer auto playing, and the file needs to wait until the Sizmek classes are loaded before proceeding, this can cause hidden elements on the stage to appear for a split second while waiting to load.

If this occurs, you can workaround this by doing the following:

Make all the elements on frame 1 have display:none

1) Give the elements you want to show on load the same class name (for example, ‘sizmek_frame_1’)
2) In the creationComplete function call for all elements with this class to display none:

   ```
   sym.$(".sizmek_frame_1").css("display","none");
   ```

3) In the StartAd() function (in the compositionReady event) give these items display:block using the following code:

   ```
   sym.$(".sizmek_frame_1").css("display","block");
   ```

FULL CODE:

Within compositionReady (please see step 2 of this guide).

```
// Sizmek function to check if Sizmek classes are initialized.
//if initialized, call startAd else add listener
function initEB(){
        if (!EB.isInitialized()) {
EB.addEventListener(EBG.EventName.EB_INITIALIZED, startAd);
        }
        else {
                startAd();
        }
}

//called once Sizmek classes are initialized
function startAd() {

//display hidden objects
sym.$(".sizmek_frame_1").css("display","block");

//play the stage/animation
sym.play();
}


//check if Sizmek is initialized
initEB();
```