MiniApp Lifecycle

Qing An, Alibaba

MiniApp Lifecycle

Draft: https://www.w3.org/TR/miniapp-lifecycle/

Overview

- MiniApp is composed of two layers: app layer and page layer.
- Therefore, MiniApp lifecycle contains **application lifecycle** and **page lifecycle**.

MiniApp Application Lifecycle Events

Initialization

- When user firstly opens a MiniApp, MiniApp will start the initialization

Running in foreground

- MiniApp initialization is completed, or when user reopens the same MiniApp that has been closed

Running in background

- When user closes the MiniApp, or go to the mobile phone’s home screen, MiniApp is running in background

Error

- MiniApp is confronted with script error

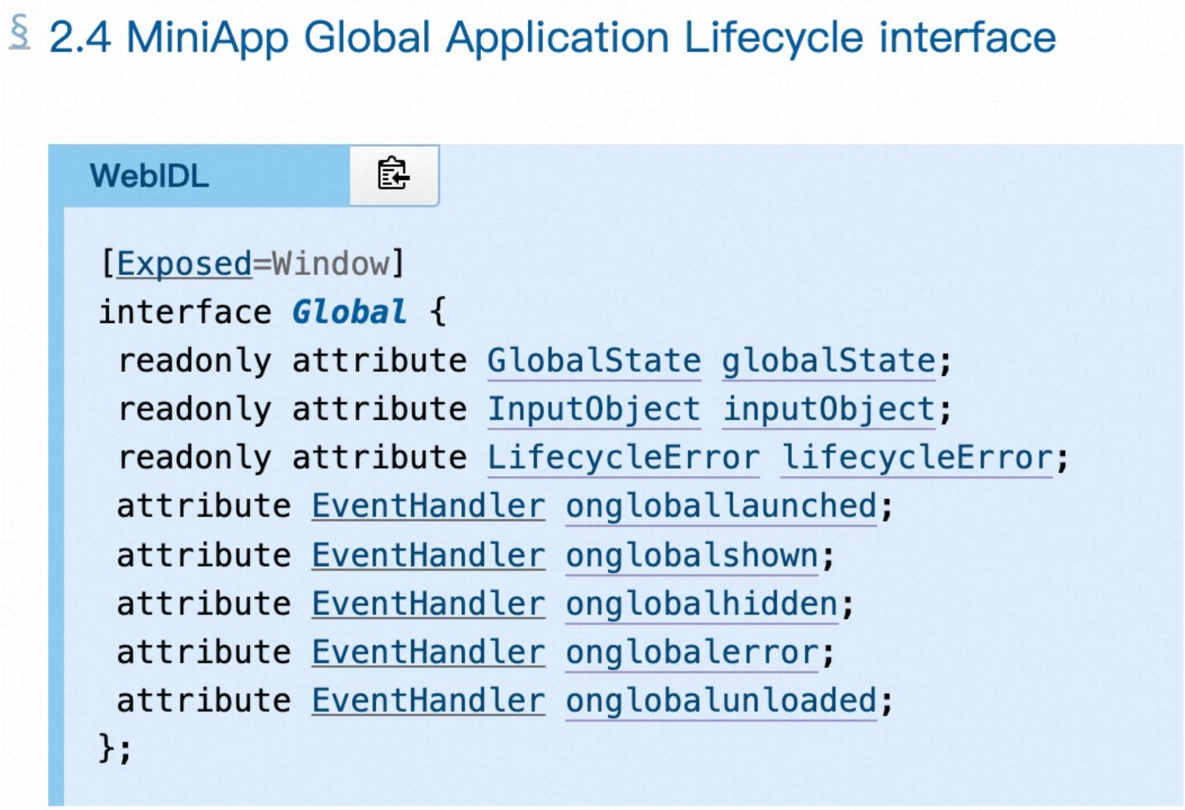
Unloading

- Mark the end of the MiniApp session, and the removal of all the temporary resources from the cache.

MiniApp Page Lifecycle Events

Loading	• MiniApp page loading procedure
First rendering ready	• MiniApp page first rendering procedure
Running in foreground	• MiniApp page in running in foreground
Running in background	• MiniApp page in running in background
Unloading	• MiniApp page is destroyed

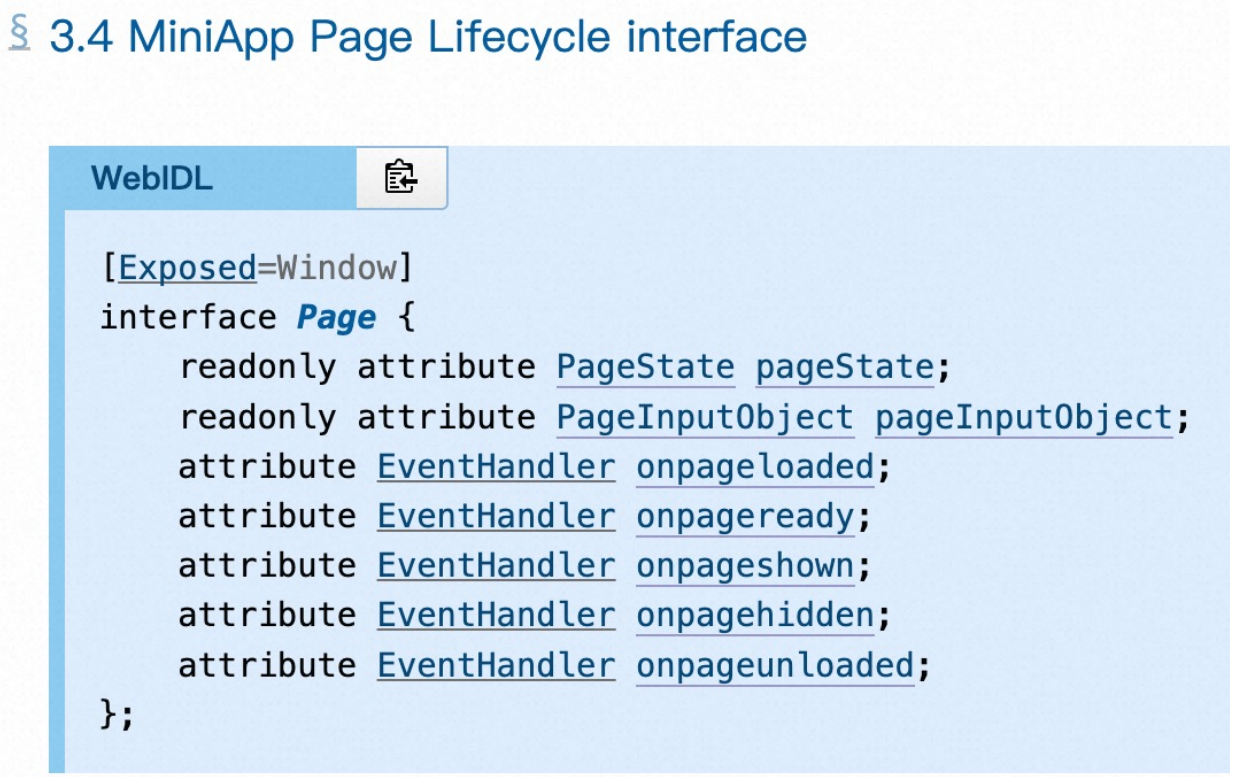
MINIAPP LIFECYCLE



§ 2.4 MiniApp Global Application Lifecycle interface

WebIDL

```
[Exposed=Window]
interface Global {
 readonly attribute GlobalState globalState;
 readonly attribute InputObject inputObject;
 readonly attribute LifecycleError lifecycleError;
 attribute EventHandler ongloballaunched;
 attribute EventHandler onglobalshown;
 attribute EventHandler onglobalhidden;
 attribute EventHandler onglobalerror;
 attribute EventHandler onglobalunloaded;
};
```

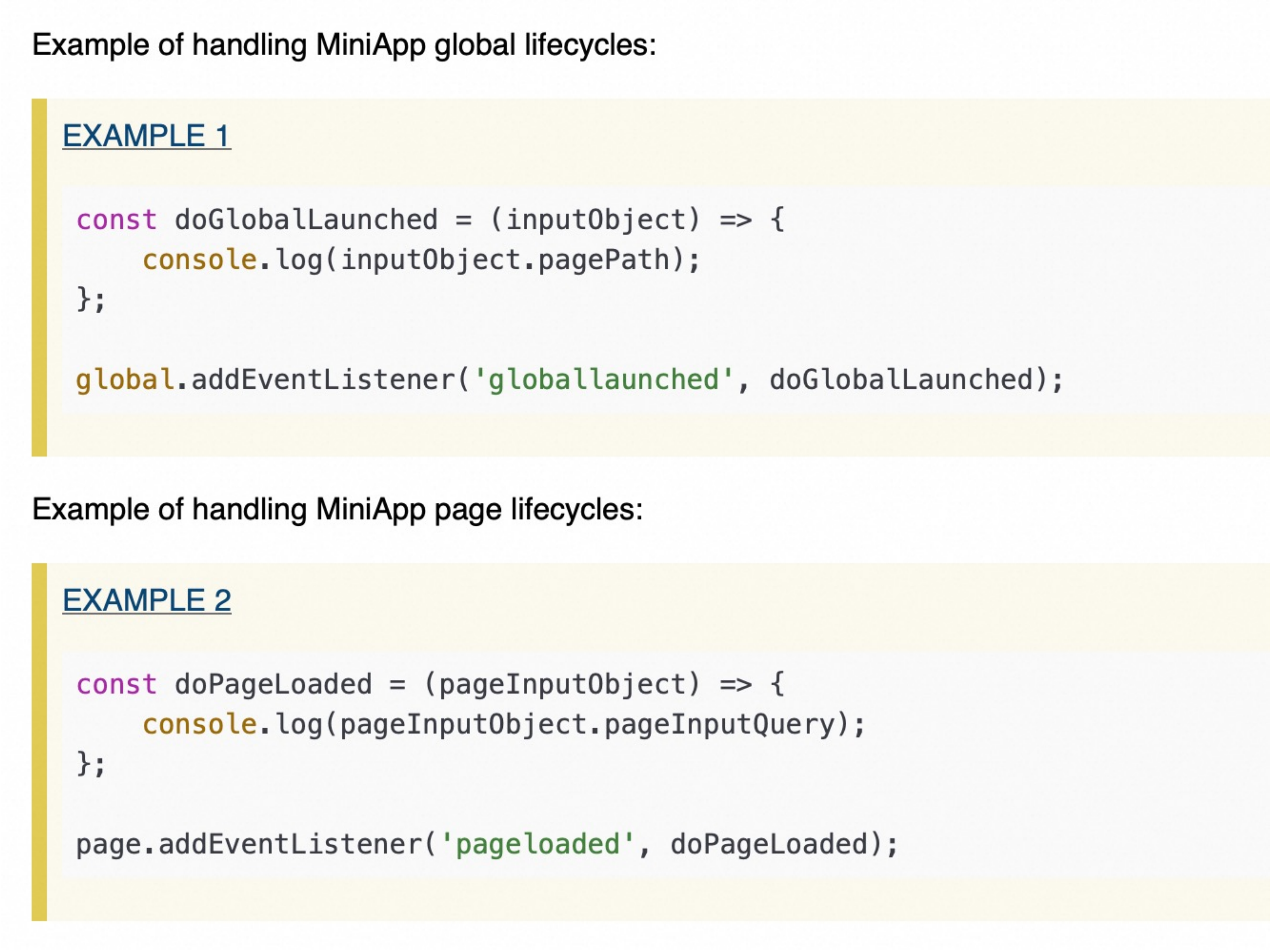


§ 3.4 MiniApp Page Lifecycle interface

WebIDL

```
[Exposed=Window]
interface Page {
    readonly attribute PageState pageState;
    readonly attribute PageInputObject pageInputObject;
    attribute EventHandler onpageloaded;
    attribute EventHandler onpageready;
    attribute EventHandler onpageshown;
    attribute EventHandler onpagehidden;
    attribute EventHandler onpageunloaded;
};
```

Sample Code



Example of handling MiniApp global lifecycles:

EXAMPLE 1

```
const doGlobalLaunched = (inputObject) => {
    console.log(inputObject.pagePath);
};

global.addEventListener('globallaunched', doGlobalLaunched);
```

Example of handling MiniApp page lifecycles:

EXAMPLE 2

```
const doPageLoaded = (pageInputObject) => {
    console.log(pageInputObject.pageInputQuery);
};

page.addEventListener('pageloaded', doPageLoaded);
```

Privacy and Security Consideration

- onShow and onHide event enables developers to know when a MiniApp is visible.
- By use of onShow event, developers can choose to process and hide the sensitive data, before MiniApp page switches to be running in foreground.
- The onUnload event provides a notification that the page is being unloaded.
- If the inputted query for the MiniApp or the inputted query for the MiniApp page contains privacy-sensitive information (e.g. user personal data), the privacy-sensitive information shall not be in cleartext.

THANK YOU!

Please join the Github issue discussion