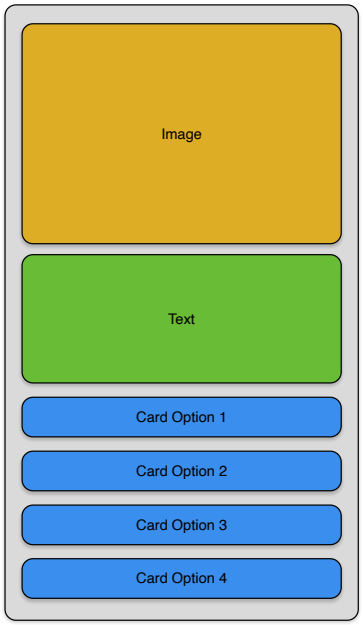
Evergreen.ink Scripting Language

Document Version 1.0.1
July 3rd, 2023

Overview:
Evergreen.ink is an interactive fiction platform developed iteratively over the past 10 years by Silverpine Software. Evergreen's primary goal is to deliver a powerful, yet easy-to-use authoring system combined with a robust reading experience tailored primarily for mobile interfaces. Evergreen offers a standalone authoring and reading experience by default, but it is also an extensible gaming platform for choice-based gaming systems.

Structure:
Evergreen stories occur in a looping fashion as the engine presents pages or "Cards" to the user and the user makes choices that determine how the story proceeds. The Card is comprised of three primary display objects:

- Image/background
- Text
- Options in the form of buttons and/or hyperlinks



Sapling:

On Show

```
var imagePath = sapling.getVariable("image");
if (sapling.getVariable("grayscale") == true)
{
    imagePath = imagePath + "?grayscale";
    if (sapling.getVariable("blur") == true)
    {
        imagePath = imagePath + "&blur";
    }
}
else if (sapling.getVariable("blur") == true)
{
    imagePath = imagePath + "?blur";
}

sapling.currentCard.image = imagePath;
```

Evergreen also contains a fairly sophisticated scripting engine called *Sapling*. *Sapling* operates in a limited JavaScript environment where authors can dynamically manipulate the visual elements of the story at runtime. *Sapling* supports complete JavaScript syntax and the Evergreen script editor implements syntax coloring as well as basic linting functions.

At a high level, Story Card objects and Card Option objects can be scriptable. By default these options are disabled, but can easily be turned on in the card editor. Once enabled, Story Cards have an *OnLoad* function that can be scriptable, and Card Options have *OnSelected* and *IsVisible* functions that can both be scripted.

Sapling maintains information about the currently presented card as well as the top level story. It also provides an interface for setting story state variables. At the top level is a globally accessible *sapling* object:

Sapling
Properties: currentCard : Card object of current card story : Story object **Functions:** setVariable(name, value) getVariable(name) : Value hasVisited(cardId) : Boolean

Setting a story state variable:

```
sapling.setVariable('likesEvergreen', true);
```

Querying a story state variable:

```
if (sapling.getVariable('likesEvergreen') == true)
{
    // Do something
}
```

Querying whether a card has already been visited:

```
if (sapling.hasVisited('1') == true)
{
    // Do something
}
```

Sapling Story Card

The sapling object allows manipulation of the visual elements through its *currentCard* property.

Card
Properties: text : String image : String options : Array of Option objects

Each of the properties will directly affect the rendered display, and are pre-loaded with whatever values were entered in the Evergreen visual editor.

Setting the text of the current card:

```
sapling.currentCard.text = 'Never gonna give you up 🎵';
```

Setting the image of the current card:

```
sapling.currentCard.image = 'https://evergreen.ink/evergreen.png';
```

Counting the number of options:

```
let numberOfOptions = sapling.currentCard.options.length;
```

Setting the text on the first option of the card:

```
sapling.currentCard.options[0].text = ‘This option is best!’;
```

Sapling Story

The story object, accessible from the global *sapling* object, contains basic, high level information about the current story:

Story
Properties: author : String description : String

Setting the text on the first option of the card:

```
let authorName = sapling.story.author;
```

Setting the text on the first option of the card:

```
let description = sapling.story.description;
```

Option IsVisible Scripting

As mentioned previously, there are three entry points for *Sapling* scripts to run:

- The Story Card *OnLoad* function
- The Card Option *IsVisible* function
- The Card Option *OnSelected* function

The *OnLoad* and *OnSelected* functions simply allow an opportunity for story state manipulation at the beginning and end of a card's display lifetime. As such, they do not need to return any value. However, the *IsVisible* script does contain a special function that Evergreen uses at runtime to determine if a particular Card Option should be rendered. (By default, all Card Options have a default value for *IsVisible* of "true".)

Story authors can manipulate this by returning a boolean (true or false) value in this function for a particular Card Option. In the example below, the Card Option's visibility will be determined by whether or not the Story Card with id '2' has been visited.

Is Visible

```
1  return sapling.hasVisited('2');
```

Disallowed JavaScript

While *Sapling* does support almost all standard JavaScript, there are a few objects and standard functions that are not permitted. The built in linter specifically disallows any asynchronous calls due to the synchronous nature of Evergreen rendering. Also disallowed are the window and navigator objects and any other browser specific functions. Evergreen stories are designed to be run in both web environments as well as native clients that don't have access to a full HTML engine.

Appendix

Sapling Data Definitions: