Affinity Space

-- Learning Metrics Integration & Development

Ruimin Zhao(rz2390) & Neng Chen(nc2734)

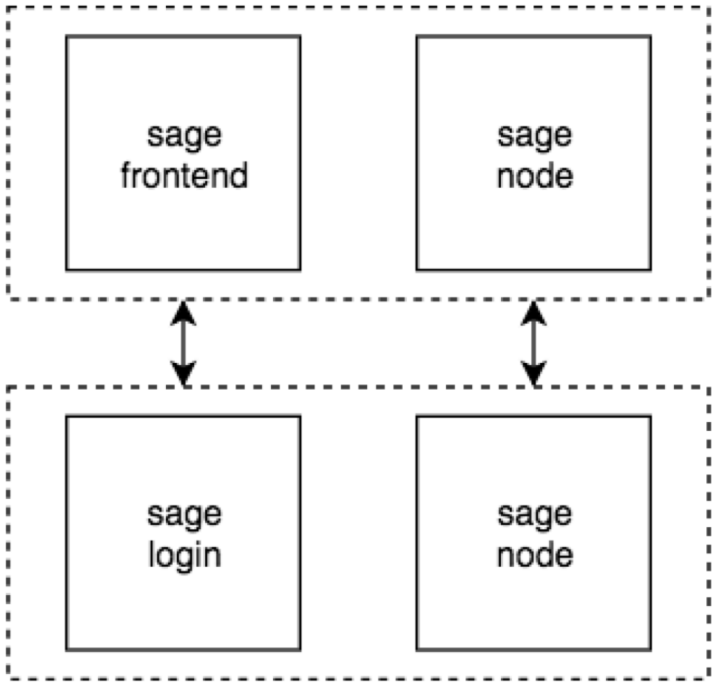
Outline

- Learning metric (sage-frontend repo & sage-login database)
- Hairball (sage-node repo & sage-node database)

Background

- 1st problem to solve: Integrate the learning metrics graph into master account without any conflicts with other features.
- 2nd problem to solve: Modify the hairball package so that it could generate game result on our sage sb2 file.
- 3rd problem to solve: Trigger the hairball assessment and store the real score for metrics graph to use (ongoing).

Architecture



Graph Reproduce

- https://gudangdaya.atlassian.net/wiki/spaces/SAG/pages/579436651/Learning+Metrics+Badges+Reproduce
- Here is the link to the detailed documentation for this portion.

Integration database

We made the metrics graphs able to be shown on master account at the first stage and then we cleared the database collections and fixed some errors (warning) in codebase to make this feature not conflicting with another feature developed by another team.


```
{
    "_id": {
        "$oid": "5acf8e30d3b2ac27f48e7380"
    },
    "name": "Class 1",
    "description": "A new class of bright students",
    "instructorId": "59f8c6fdc1bfb23c4ced8e20",
    "isDeleted": false,
    "missions": [
        "58d8476ce9a1b743936abc2e",
        "5a1ac8ca3d08f488b0a1fabd",
        "5a5881f2a55ed728d4755c9e",
        "5ab41f677be6580a68b99b2d",
        "590a2159122c012a585324d3"
    ],
    "roster": [
        "5ab2ac69dae3cc15233c2805",
        "5ae67233949dce0dcc225c64"
    ],
    "__v": 0
}
```

Hairball Modification - run sage sb2

```
        "tempo": true,
        "for each %m.varName in %s": true,
        "glide %n secs to x:%n y:%n": true
    },
    "parsonsBlocks": [],
    "children": [{
            "objName": "Sprite1",
            "scripts": [[5,
                    156.9,
                    [["F0FC6BEA-2728-F95F-5368-050373D254CE", "wait:elapsed:from:"
            "sounds": [{
                    "soundName": "pop",
                    "soundID": 0,
                    "md5": "83a9787d4cb6f3b7632b4ddfebf74367.wav",
                    "sampleCount": 258,
                    "rate": 11025,
                    "format": ""
```

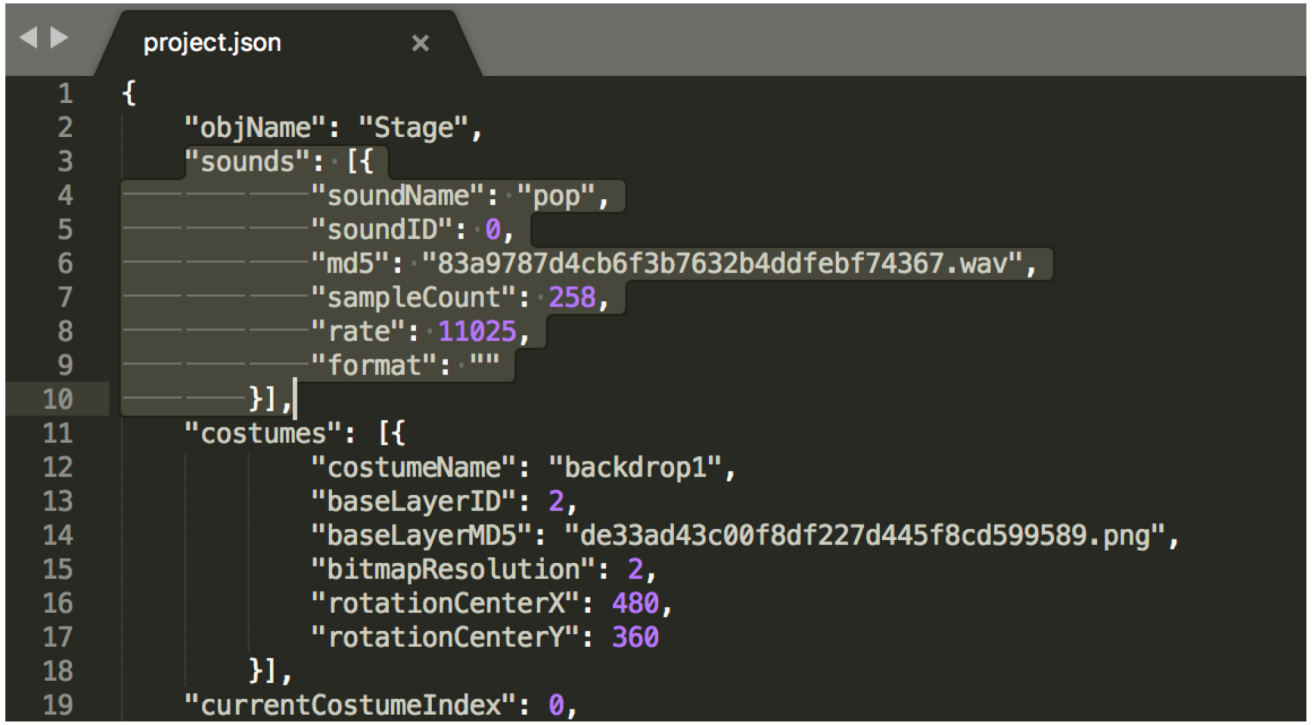
Our sage sb2 has a unique ID added to each json file

We modified Kurt library to make the Hairball able to run on our sage sb2

```
def load_block(self, block_array):
    block_array = list(block_array)
    try:
        command = block_array.pop(0)

        if command == 'procDef': # CustomBlockType definition
            spec = block_array[0]
            return kurt.Block('procDef', self.custom_blocks[spec])

        if command == 'call': # CustomBlockType call
            block_type = self.custom_blocks[block_array.pop(0)]
        else:
            block_type = kurt.BlockType.get(command)
    except kurt.UnknownBlock:
        command = block_array.pop(0)

        if command == 'procDef': # CustomBlockType definition
            spec = block_array[0]
            return kurt.Block('procDef', self.custom_blocks[spec])

        if command == 'call': # CustomBlockType call
            block_type = self.custom_blocks[block_array.pop(0)]
        else:
            block_type = kurt.BlockType.get(command)
```

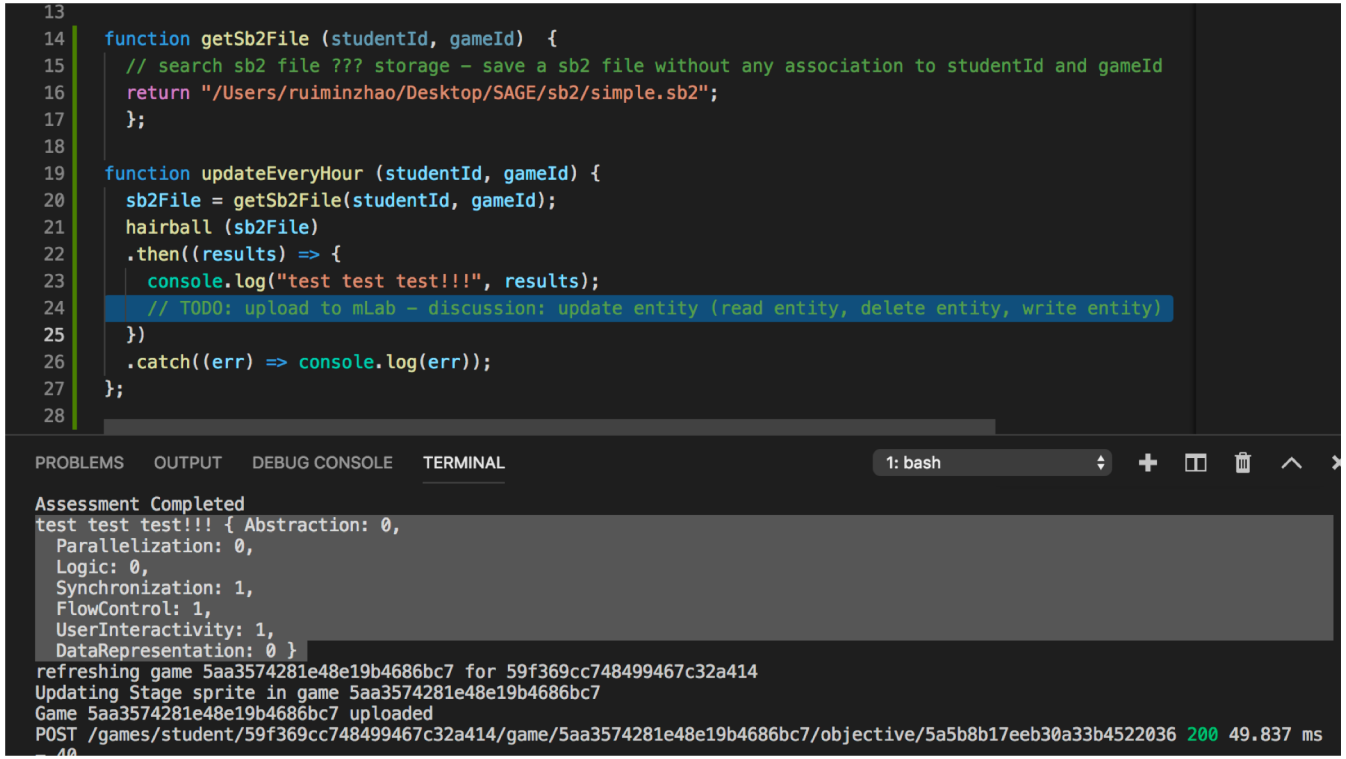
Hairball Modification - run json

After managed to make the Hairball run on sage sb2 file, we then tried to optimize this pipeline further - to avoid keeping all the redundant files such as images and sound files generated all together with that core json file. This is for the purpose of speeding up the frequent uploading and downloading process of that sb2 file.



Assessment Trigger

After we modified the Hairball related libraries and make it able to get real score, we then started to find the correct way to trigger the assessment process so as to get that real score in program and then save them into metric collection so that our learning metrics graphs could be generated based on real score from real assessment rather than being hardcoded.



Future Work

- Finalize the way we store and assess a game from both student and teacher's perspective, and finish the database communication portion of code.
- Create more metrics graph based on different use of the hairball assessment result of a game in the front-end UI.

Q&A

- Thank You!