

User Interface Options Version A Technical Considerations

 This is an old working page - the technical strategy has evolved since.

Interface Customization: A Possible View of the World

- fluid comes with several skins
- each skin has predefined parts (aspects) - these are analogous to the options available when customizing the user interface
- customization would allow the user to select from the predefined aspects of different skins
- the user would then essentially have their own custom skin
- new skins could also be introduced by the integrator, provided that the aspects are communicated to the preferences tool
- styleable would be included on every page
- once the user has saved preferences, styleable could query the server and restyle the page as necessary

Questions to answer

How do we deliver the style sheet? Can we get the results we want without using !important?

Eli is currently building test cases to explore this area. One important thing to clarify when it comes to getting the intended result is called [CSS Specificity](#)

What are the aspects of a skin?

This will be determined in conjunction with Gary and Jacob. Gary is currently working on the design which will determine how fine grained these 'aspects' are. Given Jacob's work on the fluid skins, he will be able to give context around where the natural fault lines in a skin are.

Some possibilities for what might be included in a skin:

- Font face
- Font size
- Text spacing
- Color scheme

How do we represent skins and aspects?

Fragments of CSS - one option is to have each part of the theme specified separately. Possibilities for this:

- in individual files for example "mist-color-scheme.css"
- in a JSON structure

```
{"color-scheme": [{"mist": [<array of CSS snippets>]}]}
```

One problem with this strategy is that it doesn't play to normal work-flow. Ordinarily, CSS developers don't fragment their style sheet so much, and they certainly don't put it into a JavaScript object.

Another possibility is to add annotations in the CSS file that could be preprocessed into a javascript object. This would allow the CSS developer to build a single style sheet for a skin but would introduce the need for the developer to add the annotations to the file. This might look like:

```
/*@color-scheme*/  
color: gray;  
background-color: white
```

What strategy would Styleable employ?

Here is one scenario of how styleable would work:

1. Ask the server: are the correct styles already here? If yes we're done, if no continue.
2. Ask the server: do you have a style sheet for me? If yes inject it, if no continue.
3. Ask the server: do you have preferences for me? If yes generate and inject the style sheet.