



boostache



boost ache

```
<h1>{{header}}</h1>
```

```
{{#bug}}
```

```
{{/bug}}
```

```
{{#items}}
```

```
  {{#first}}
```

```
    <li><strong>{{name}}</strong></li>
```

```
  {{/first}}
```

```
  {{#link}}
```

```
    <li><a href="{{url}}">{{name}}</a></li>
```

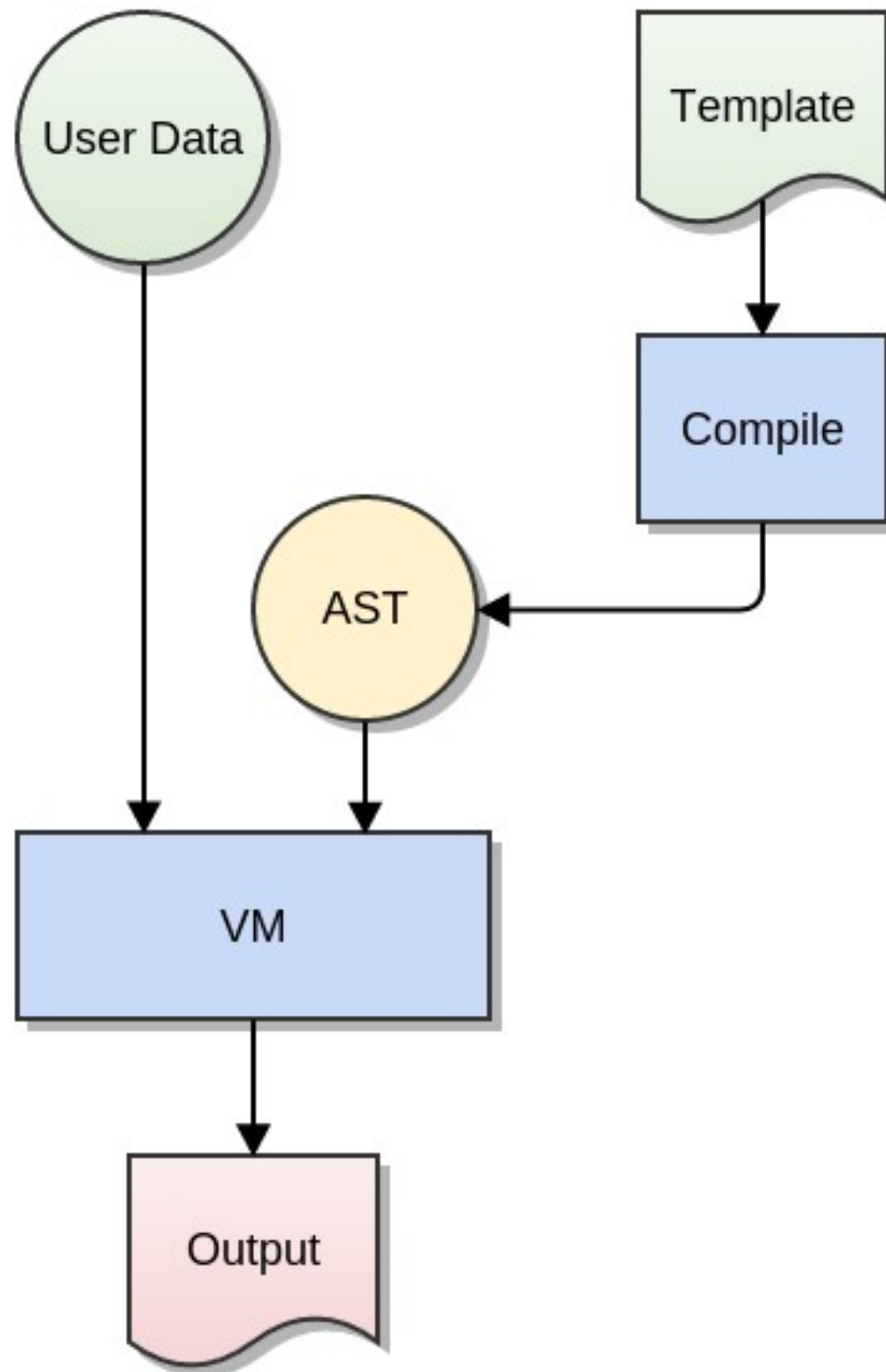
```
  {{/link}}
```

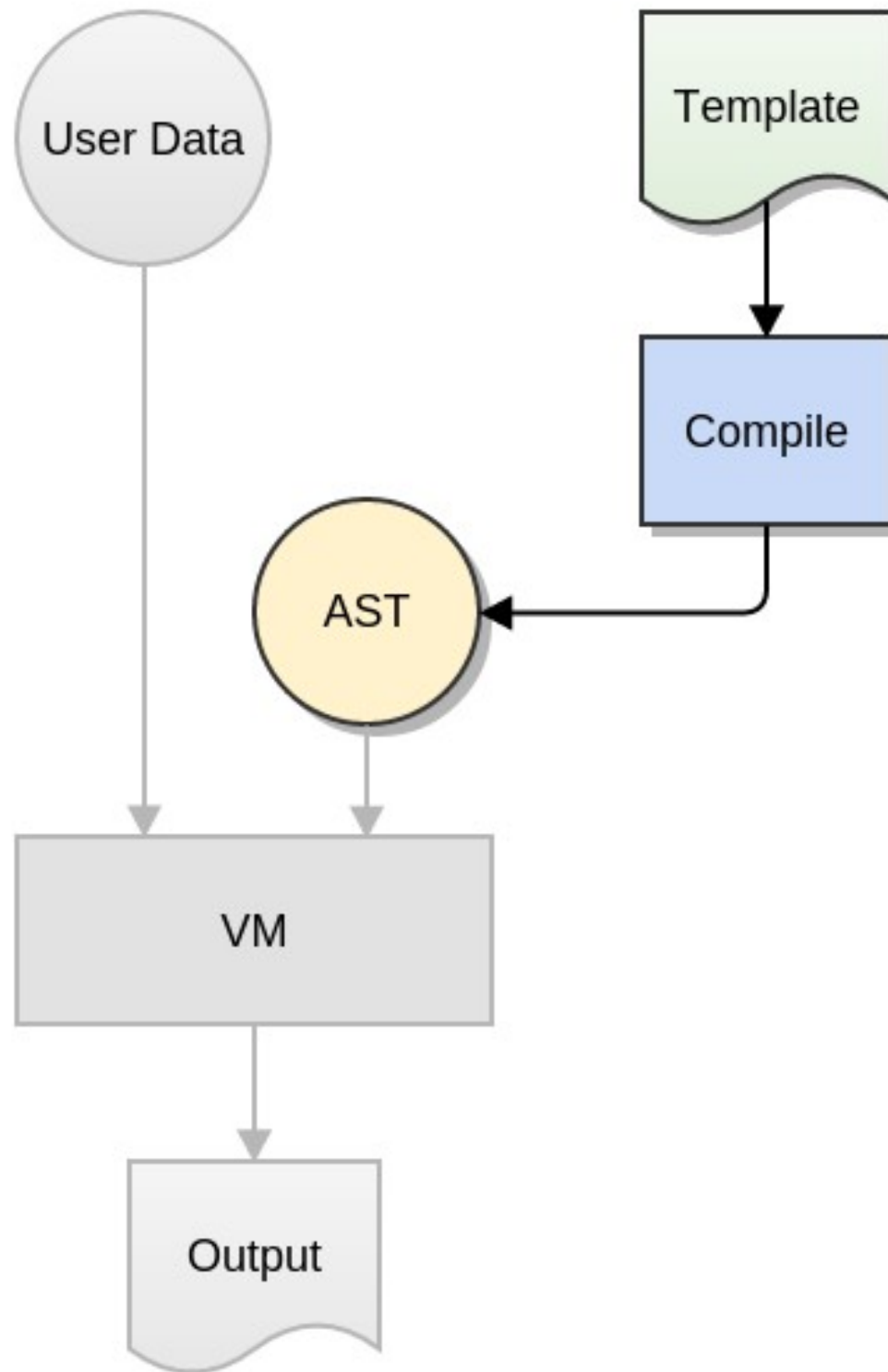
```
{{/items}}
```

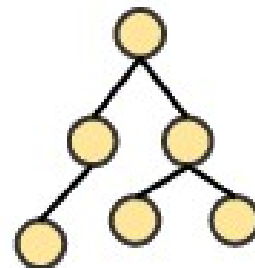
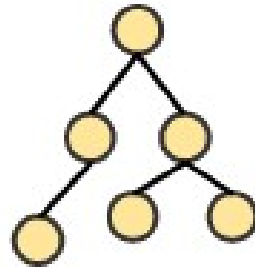
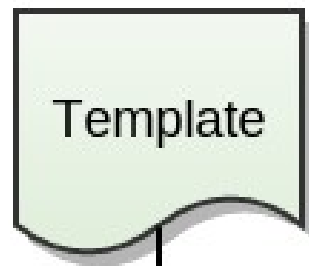
```
{{#empty}}
```

```
  <p>The list is empty.</p>
```

```
{{/empty}}
```



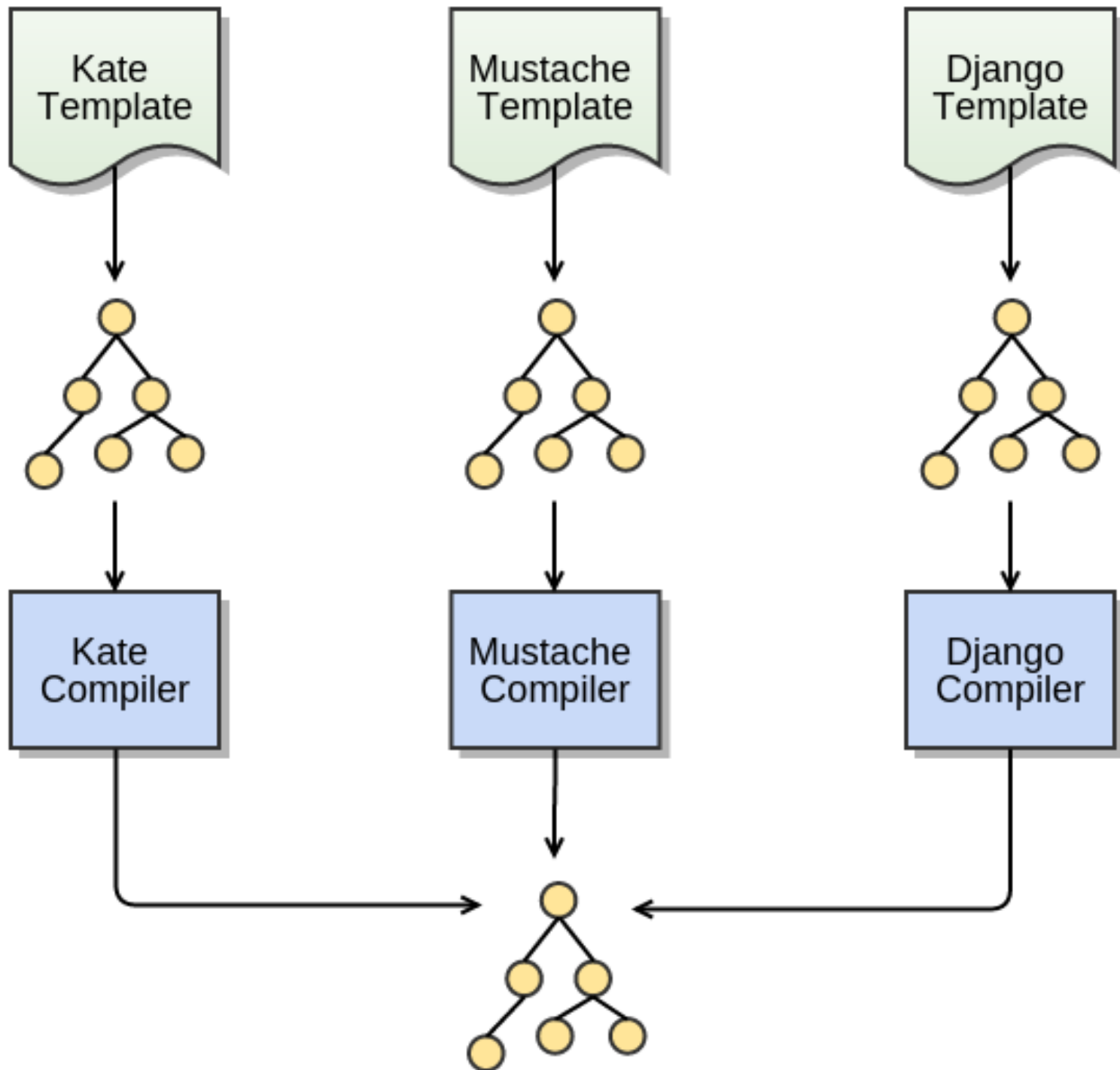


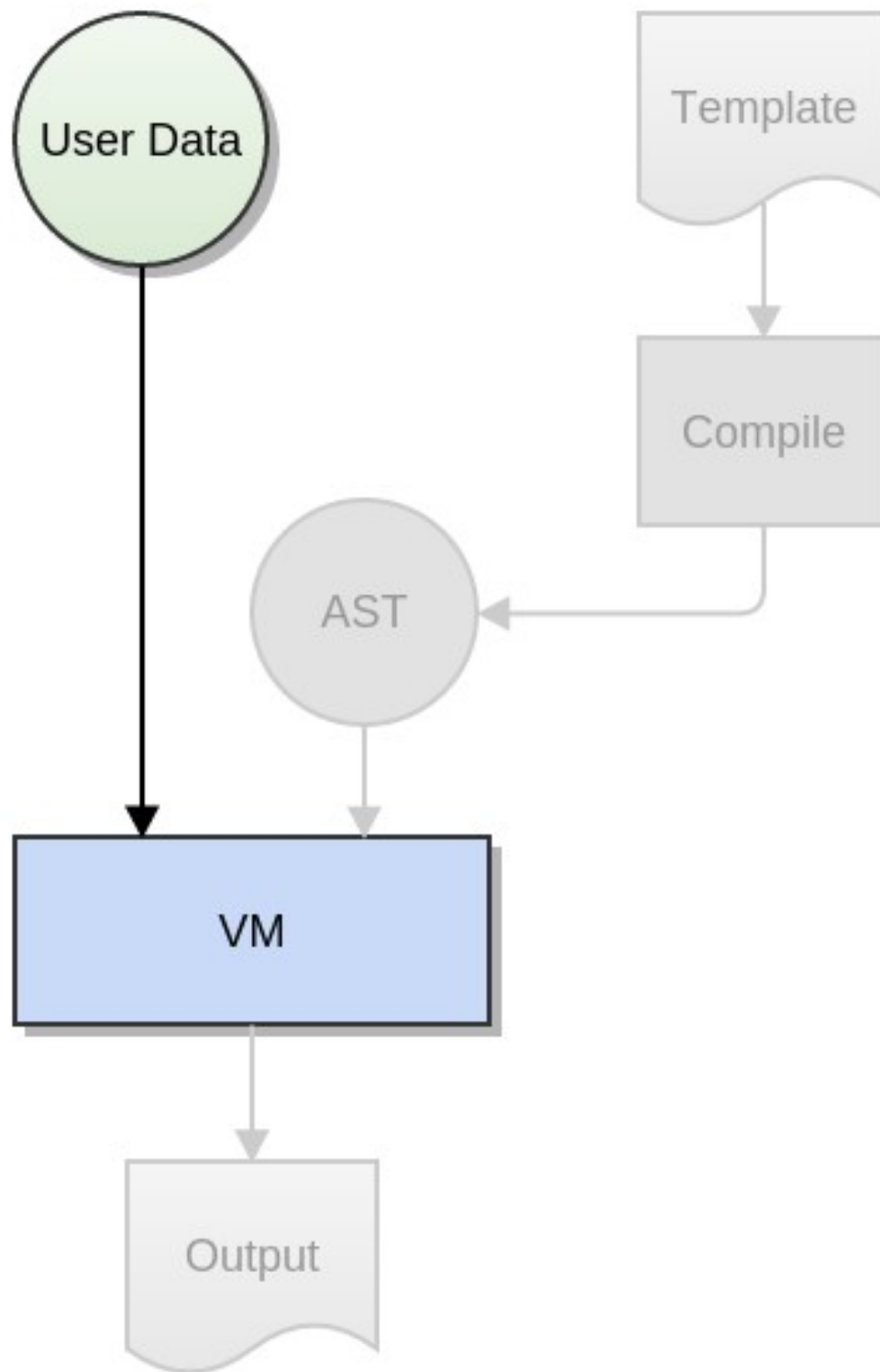


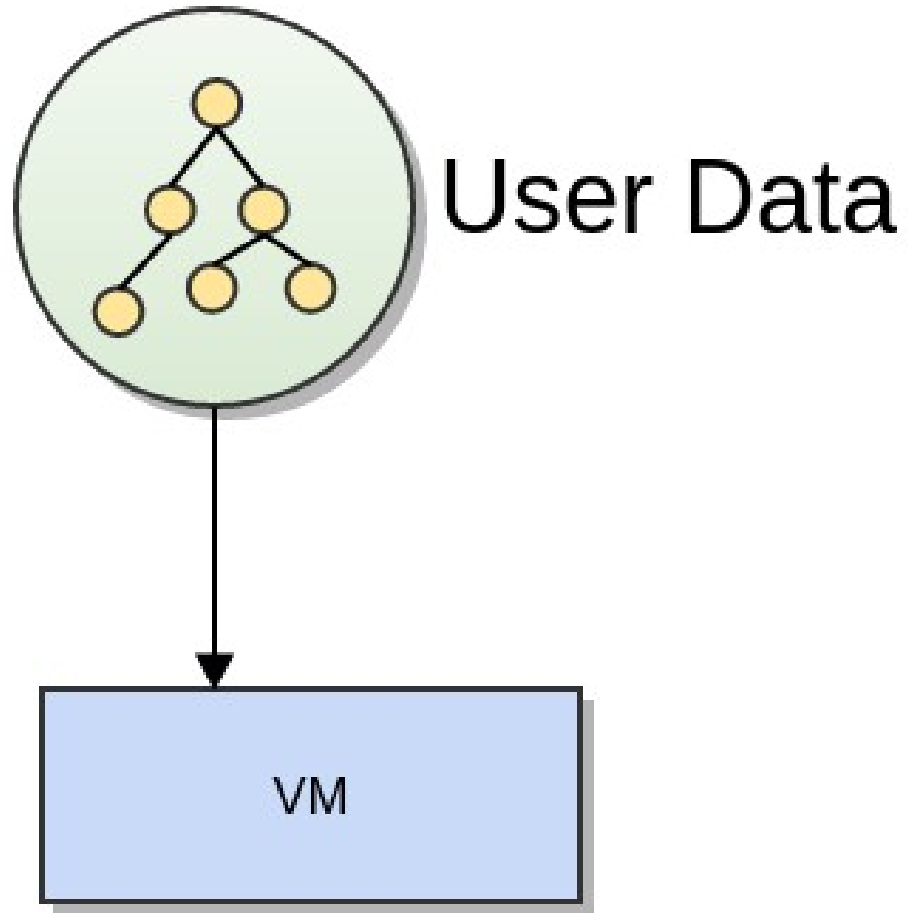
Boost.Spirit

Modern
Techniques

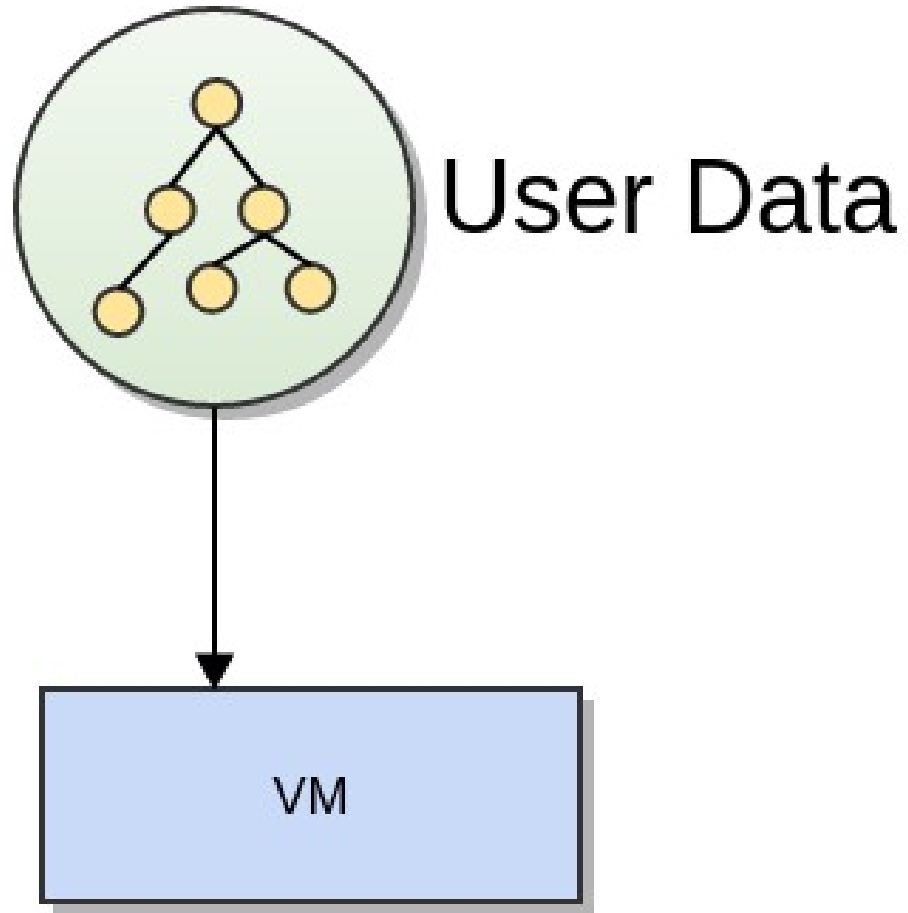
engine
primitive AST







What structure should the User Data be in?



JSON?

Variant recursive data
structure?

Maps? Vectors?

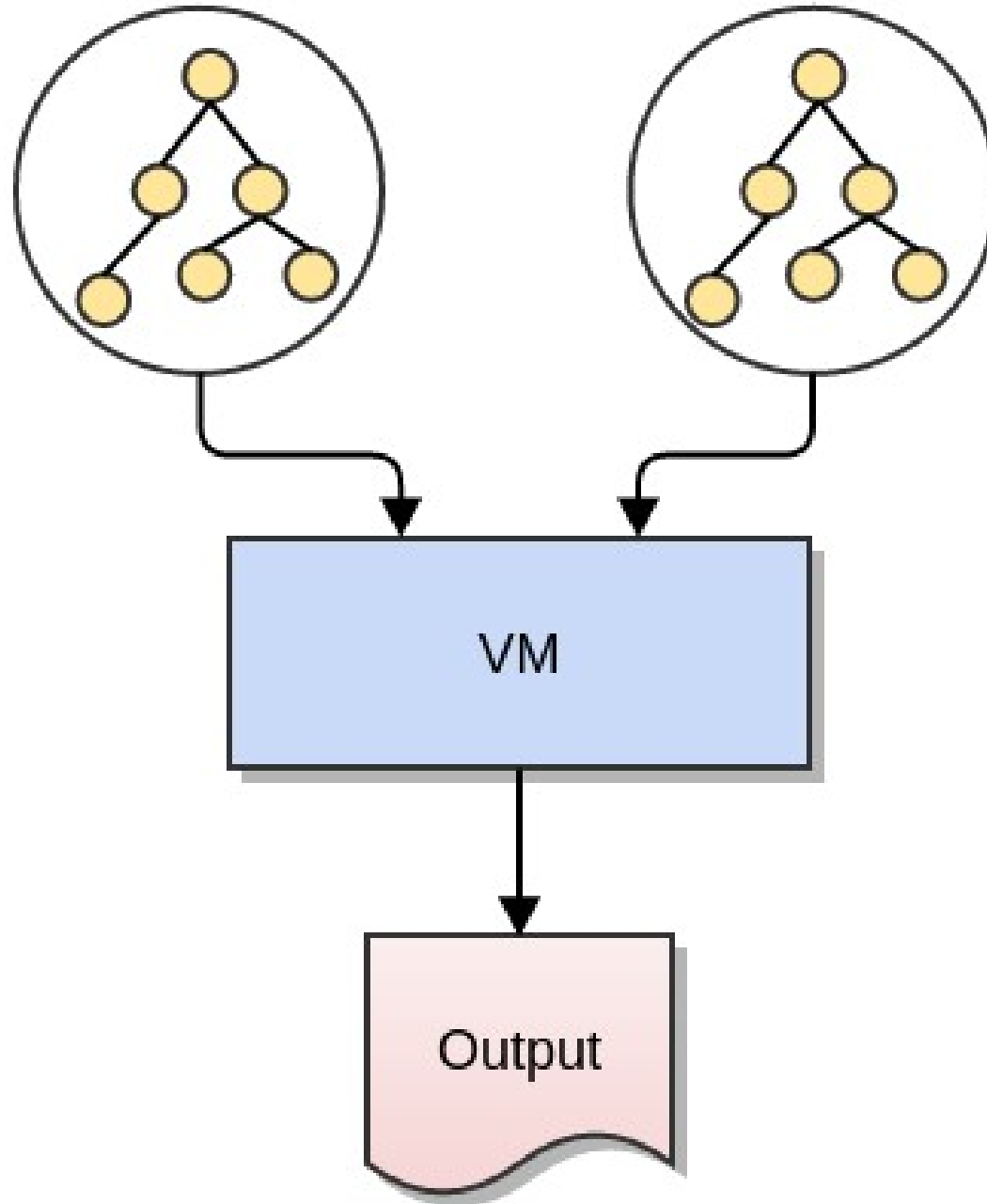
```
struct my_type;
using my_object_t = std::map<std::string,my_type>;
using my_list_t   = std::list<my_type>;

struct my_type : boost::spirit::extended_variant<
    bool
    , std::string
    , int
    , my_object_t
    , my_list_t
>
{
    my_type() : base_type() {}
    my_type(bool rhs) : base_type(rhs) {}
    my_type(std::string const & rhs) : base_type(rhs) {}
    my_type(const char * rhs) : base_type(std::string(rhs)) {}
    my_type(int rhs) : base_type(rhs) {}
    my_type(my_object_t const & rhs) : base_type(rhs) {}
    my_type(my_list_t const & rhs) : base_type(rhs) {}
};
```

```
my_object_t data = {
    {"name"           , "Jeff"},
    {"escaped_name"   , "<h1>Jeff</h1>"},
    {"whoot"          , "yipee"},
    {"bar"             , false},
    {"foo"             , my_list_t {
        my_object_t{ {"whoot" , "yipee 1"}}
        , my_object_t{ {"whoot" , "yipee 2"},
                        {"bar"    , true}}
        , my_object_t{ {"whoot" , "yipee 3"} } }}
};
```

Data AST

Instruction AST





`ciere labs.org`

`@ciere labs`

`github.com/ciere labs`