Text File Format for Simple UML Diagrams

This document defines the syntactic format of text files that store Simple UML diagrams. The file contains zero or more definitions for the three kind of diagram element: pacakge, class, and operation. The order of the elements within the file is irrelevant, since each element is defined with all the information needed to render it in a display.

The syntactic format for each element parallels the Java API specification of Simple UML diagrams. This API spec is in the ../java_api directory. The comments in the API spec have further explantory details about structure and display of Simple UML diagrams.

Here is the syntax:

```
page: 1
  package:
    name: <identifier>
    name_prefix: <string>
    name_suffix: <string>
    height: <int>
    width: <int>
    x_coord: <int>
    y_coord: <int>
    components: <element_name> ...
    show_as_contained: <boolean>

  class:
    name: <identifier>
    name_prefix: <string>
    name_suffix: <string>
    height: <int>
    width: <int>
    x_coord: <int>
    y_coord: <int>
    inherits_from: <class_name> ...
    attributes: <name_type_pair> ...
    strong_aggregates: <name_type_pair> ...
    weak_aggregates: <name_type_pair> ...
    reference_aggregates: <name_type_pair> ...
    union_aggregates: <name_type_pair> ...
    operation_attributes: <operation_signature>
    operation_aggregates: <operation_signature>

  operation:
    name: <identifier>
    name_prefix: <string>
    name_suffix: <string>
    height: <int>
    width: <int>
    x_coord: <int>
    y_coord: <int>
    inputs: <name_type_pair> ...
    outputs: <name_type_pair> ...

page: 2

  ...

page: n
```

Here are defintions of non-terminals used above:

- An <identifier> follows the rules of programming language names, i.e., starts with a letter, followed by letters, digits, underscores.
- An <element_name> is the identifier name of a diagram element, i.e., a package, class, or operation.
- A <class_name> is the identifier name of a class defined in the diagram.
- A <name_type_pair> is a colon-separated pair of identifiers. The first item in the pair is any legal indentifier. The second item in the pair is the name of a class defined in the diagram. The <name_type_pair> can be optionally preceded with a UML visibility character for public, private, or protected.
- An <operation_signature> has the format `<identifier>>(<inputs>)[:(<outputs>)]`. The <identifier> is the name of an operation defined in the diagram. The <inputs> and <outputs> are comma-separated <name_type_pairs>.