Concrete Codecharts

Codecharts [1] are a formal visual language for specifying the structure of object-oriented programs, frameworks and design patterns. Their vocabulary represents common concepts in statically typed class based object-oriented program design: classes, methods, signatures, properties and relationships.

Abstract Codecharts

Concrete codecharts contain a lot of information that is not relevant to its semantics, e.g. size and layout. We prefer, therefore, to reason over an abstract syntax that discards such semantically unimportant geometric information.

An abstract codechart is a tuple composed of multisets (bags) containing labels and a function preserving semantically desirable topological information. Bags capture occurrences of labels such as to reason over them independently.

Abstract Codecharts

Programs

The semantics of codecharts are based on OO programming languages and therefore requires a mapping to OO concepts be defined. For example, to map the class name Media to the Java class Media, Abstract to the keyword abstract, and Inherit to the keywords extends and implements.

Other OO languages can be used, such as C++ (below), if such a mapping can be defined for it. For example, Media is Abstract as it contains a pure virtual method.

Syntax Reference

In [2] we formalized this concrete syntax, thereby providing means to decide the syntactic well-formedness of codecharts.

In the above, class Media is Abstract, classes Music and Film both Inherit from Media, and these classes all have a method with the signature getTitle. The below specifies the same design decisions except that Media is not explicitly specified as Abstract.

Future work includes defining a model theoretic semantics and to investigate applications such as visualizing program metrics.

References

Codecharts: Roadmaps and Blueprints for OO Programs.

Formalizing the syntax of codecharts.
In Proc. of DMS 2012 (VLC 2012), Miami, USA, August 2012.