

Type systems of programming languages generally do not understand external data. In statically typed programming languages, developers handle data by writing typed interfaces and data access code by hand or using external code generation tools. Type providers, originally developed for F#, are an alternative, more integrated approach to code generation that improves compile-time safety and developer experience by inferring types from data samples.

This thesis explores the implementation of type providers in TypeScript. We propose a design for the feature and implement a functional prototype as an extension of the current TypeScript compiler. We demonstrate the feature's usefulness by implementing type provider packages for CSV, XML and JSON.