The equality reflection rule states that propositionally equal terms are judgmentally equal. It is generally frowned upon, as it makes type checking undecidable. Nevertheless, it would be useful to have a workable implementation of a type theory with equality reflection, for instance to tackle Voevodsky’s Homotopy Type System. The difficulties created by equality reflection are numerous. It destroys the structural rules of exchange and strengthening, $\eta$-reductions cannot be performed without explicit typing annotations in the terms, and injectivity of type constructors becomes an unreasonable expectation. In the talk, I shall discuss the design and implementation of a type theory with reflection that works around these complications while still being a useful tool.