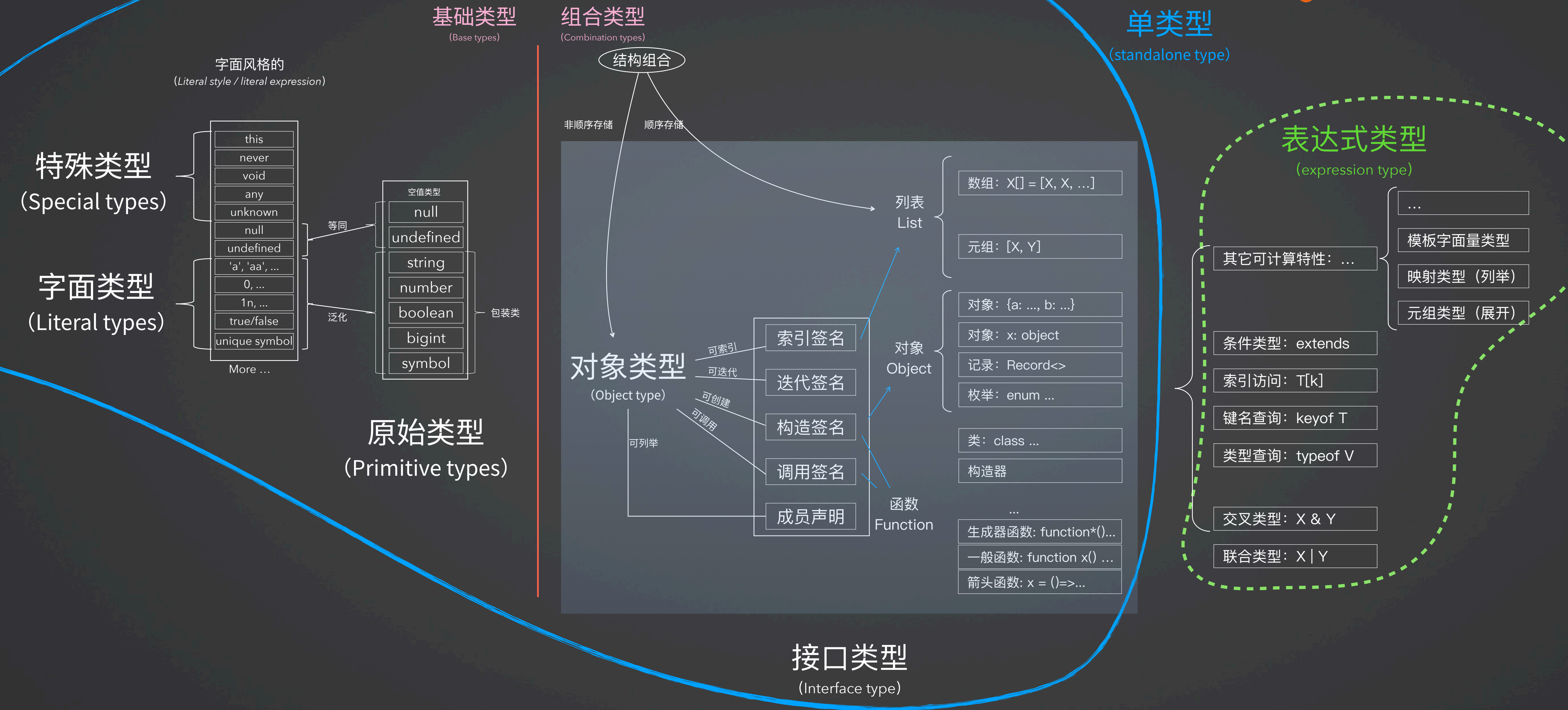


33 | 泛型工具实践1 – 识别与推断

周爱民 (Aimingoo)

目录

- 1 理解 JS / TS 中的某些类型的基础概念
- 2 单类型的识别
- 3 利用推断的类型识别
- 4 等值检测 (Equal)
- 5 总结



简单类型间的赋值兼容性

x \ T	null	undefined	void	never	any	unknown	'a'	1	true	false	string
T_null	TRUE	FALSE	FALSE	TRUE	boolean	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
T_undefined	FALSE	TRUE	FALSE	TRUE	boolean	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
T_void	FALSE	TRUE	TRUE	TRUE	boolean	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
never	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
any	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
unknown	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
'a'	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE
1	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
T_true	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
T_false	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE
string	FALSE	FALSE	FALSE	TRUE	boolean	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE

- * never 可以赋给任何类型，但只能被 any 和 never 赋值
- * unknown 只能赋给 any 和 unknown，但可以被任何类型赋值

总结

1. 谓词签名（以及断言签名）与它的类型

2. `IsXXX<>` 中的处理

- ▶ any / never 和 联合类型的特殊处理
- ▶ 充分利用类型（泛型声明可以反向推断类型）

3. 没有可信赖的 `Equal<A, B>` 泛型工具

- ▶ 某些情况下需要有选择性的使用

THANKS