

7.2.15 Abstract Equality Comparison

1. If $\text{Type}(x)$ is the same as $\text{Type}(y)$, then
 - a. Return the result of performing **Strict Equality Comparison** $x === y$.
2. If x is **null** and y is **undefined**, return **true**.
3. If x is **undefined** and y is **null**, return **true**.
- ...
10. If $\text{Type}(x)$ is either String, Number, BigInt, or Symbol and $\text{Type}(y)$ is Object, return the result of the comparison $x == \text{ToPrimitive}(y)$.
11. If $\text{Type}(x)$ is Object and $\text{Type}(y)$ is either String, Number, BigInt, or Symbol, return the result of the comparison $\text{ToPrimitive}(x) == y$.
- ...

MemberExpression :

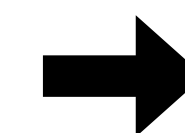
PrimaryExpression

MemberExpression [Expression]

MemberExpression . IdentifierName

new *MemberExpression Arguments*

new	<i>MemberExpression</i>	<i>Arguments</i>
new	x	()
new	x	(x)
	(the shortest string of <i>MemberExpression</i>)	(...x)
		(x,)

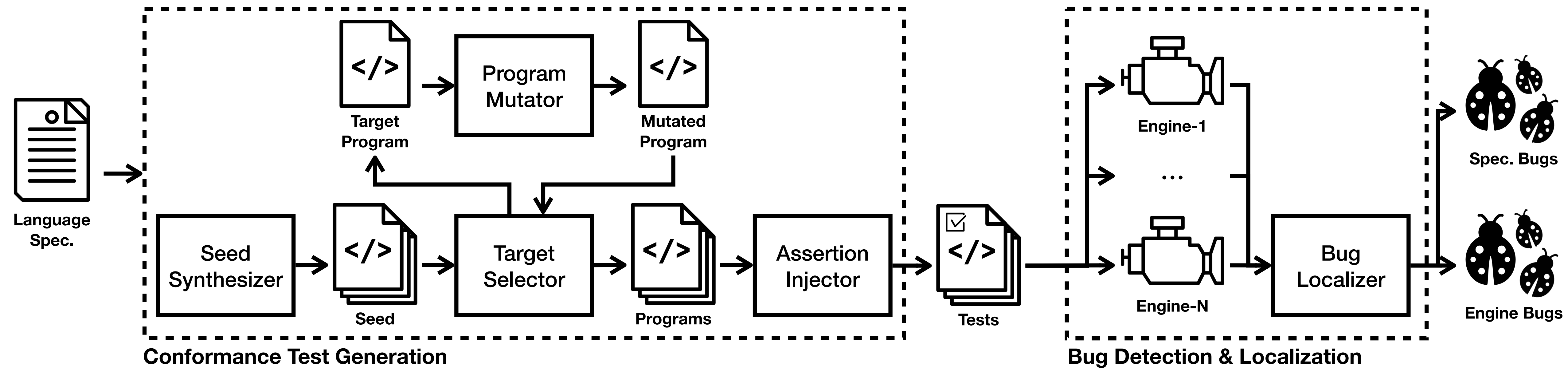


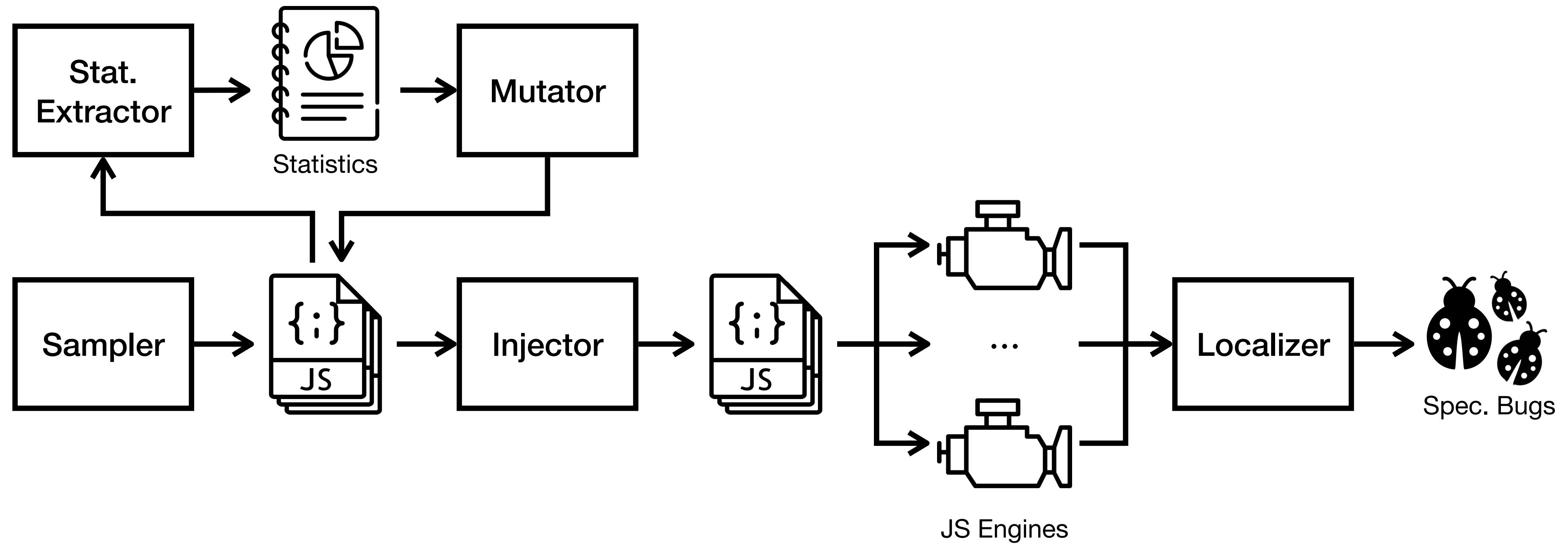
```

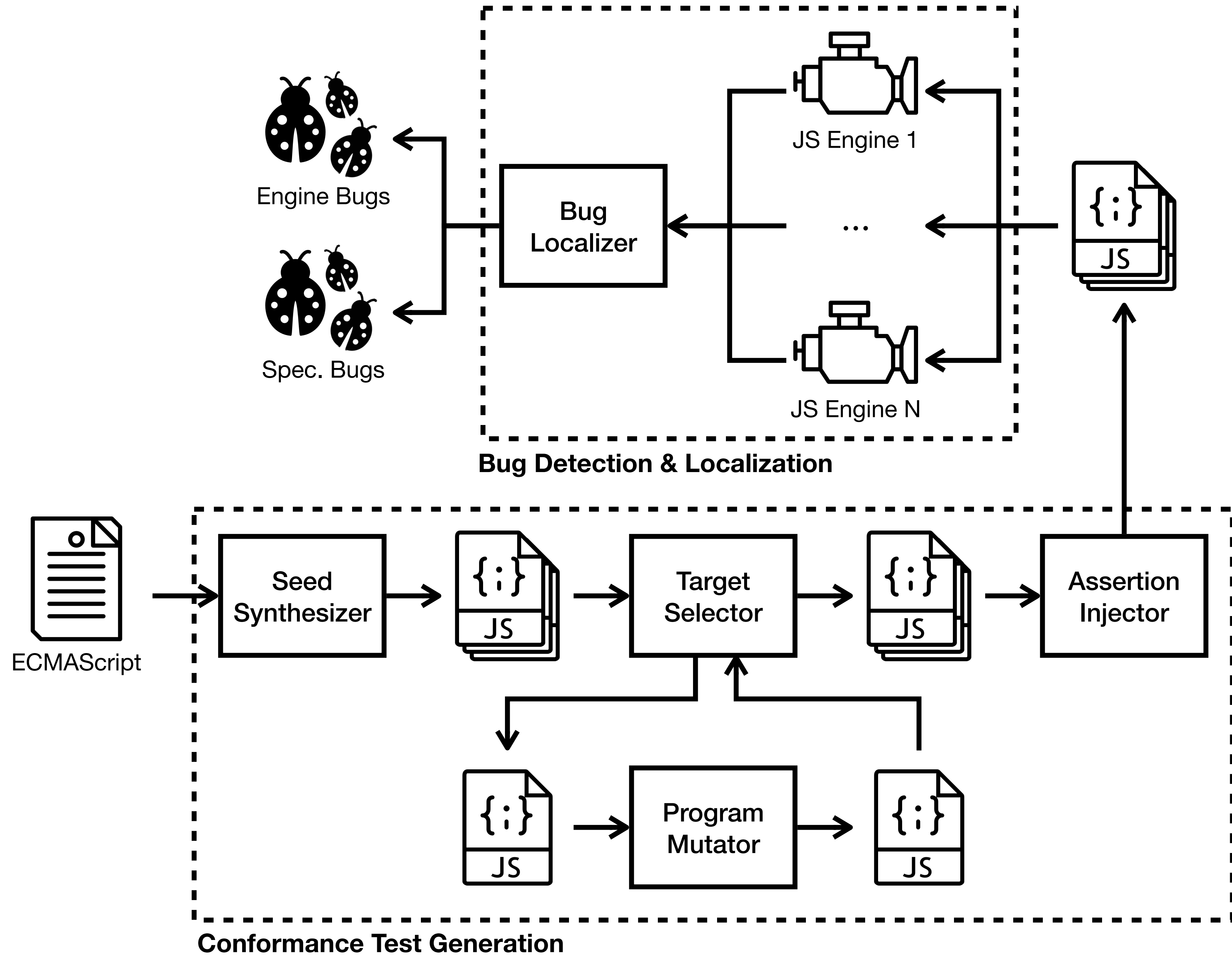
new x()
new x(x)
new x(...x)
new x(x,)

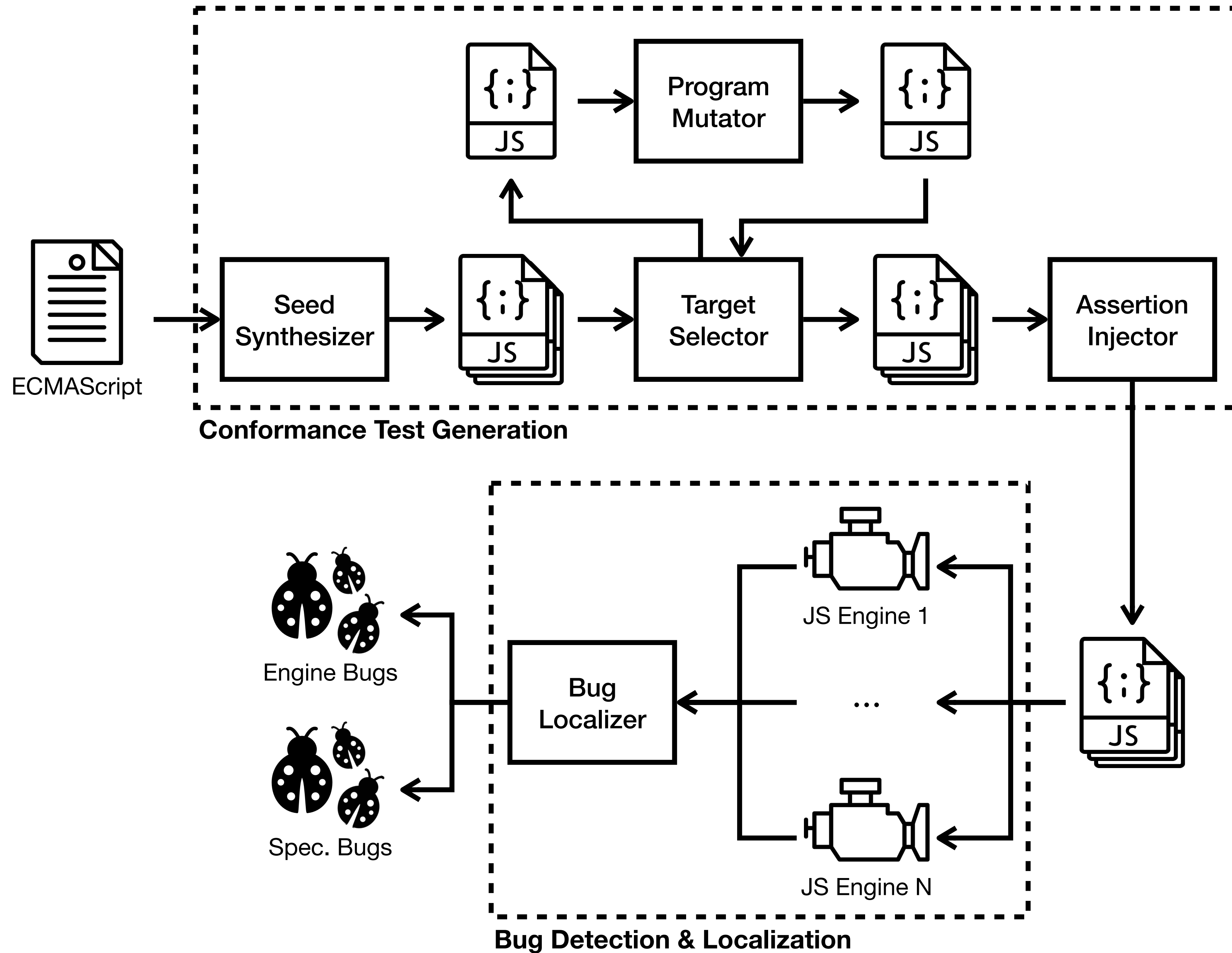
```

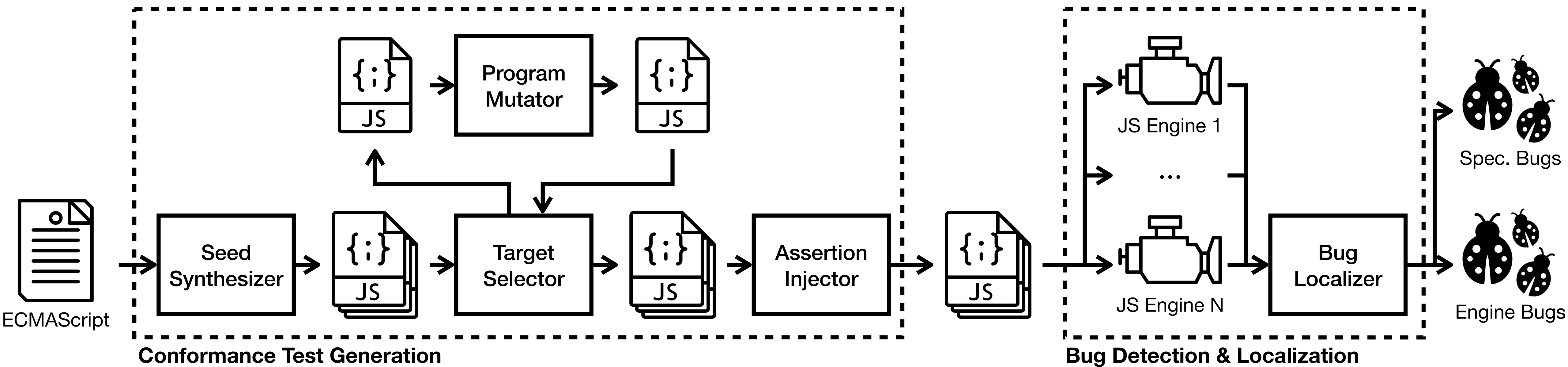
Array.prototype.indexOf (*searchElement* [, *fromIndex*])











*ArrayLiteral*_[Yield, Await] :

- [*Elision*_{opt}]
- [*ElementList*_[?Yield, ?Await]]
- [*ElementList*_[?Yield, ?Await] , *Elision*_{opt}]