A method to filter the receiver against some condition

I frequently see code that uses some value if that value satisfies a certain condition, and something else otherwise.

```
a.some_condition ? a : b
```

And in most cases, the value of `a` is non-nil when the condition is satisfied.

I propose to have a method, perhaps named `verify`, which would be implemented to be equivalent to this definition:

```
class Object
  def verify
    self if yield(self)
  end
end
```

Then, we can write the expression above (assuming `a` is non-nil when the condition is satisfied) like this:

```
a.verify{|a| a.some_condition} || b
```

Perhaps it would also be useful to do something like:

```
a.verify{|a| a.some_condition}.chaining_of_more_methods
```

I have no particular pro or con opinion about your proposal, but I just want to say that the last variant is not very pretty.

The lonely guy operator staring at the dot before him is pressed hard against the wall behind him there, the "}" character!

I don't see any improvement. The new way that would be possible with the `verify` method is longer and more complicated than the simple and straightforward `a.some_condition ? a : b`.

It seems useless without method chain, i.e., variable `a` is too simple as an example.

That is, this is a variant of `yield_self`.

```
foo.bar.zot.yield_self {|a| a.some_condition ? a : b}
```

or

```
foo.bar.zot.verify {|a| a.some_condition} || b
foo.bar.zot.verify(&:some_condition) || b
```

But `verify` is long and sounds ambiguous.

This came to my mind, but looks magical a little.

```
foo.bar.zot.if?(&:some_condition) || b
```
#4 - 01/23/2019 09:18 AM - sawa (Tsuyoshi Sawada)
I came up with an idea better than this (https://bugs.ruby-lang.org/issues/15557).
So I withdraw this proposal. Please close it.

#5 - 01/23/2019 09:33 AM - nagachika (Tomoyuki Chikanaga)
- Status changed from Open to Closed

#6 - 01/23/2019 09:34 AM - nagachika (Tomoyuki Chikanaga)
- Related to Feature #15557: A new class that stores a condition and the previous receiver added