Funcons-beta: Booleans

The PLanCompS Project

`Funcons-beta/Values/Primitive/Booleans/Booleans.cbs`[*]

Booleans

```
[ Datatype booleans
     Alias bools
    Funcon true
    Funcon false
    Funcon not
    Funcon implies
    Funcon and
    Funcon or
    Funcon exclusive-or
     Alias xor ]
```

```
Datatype booleans ::= true
                    | false
```

```
Alias bools = booleans
```

```
Funcon not(_ : booleans) : ⇒ booleans
```

not(B) is logical negation.

```
Rule not(false) ⇝ true
Rule not(true) ⇝ false
```

[*]Suggestions for improvement: `plancomps@gmail.com`.
Issues: `https://github.com/plancomps/CBS-beta/issues`.

Funcon implies(_ : booleans, _ : booleans) : $\Rightarrow$ booleans

implies(B_1, B_2) is logical implication.

Rule implies(false, false) $\rightsquigarrow$ true
Rule implies(false, true) $\rightsquigarrow$ true
Rule implies(true, true) $\rightsquigarrow$ true
Rule implies(true, false) $\rightsquigarrow$ false

Funcon and(_ : booleans*) : $\Rightarrow$ booleans

and($B, \cdots$) is logical conjunction of any number of Boolean values.

Rule and( ) $\rightsquigarrow$ true
Rule and(false, _* : booleans*) $\rightsquigarrow$ false
Rule and(true, B^* : booleans*) $\rightsquigarrow$ and(B^*)

Funcon or(_ : booleans*) : $\Rightarrow$ booleans

or($B, \cdots$) is logical disjunction of any number of Boolean values.

Rule or( ) $\rightsquigarrow$ false
Rule or(true, _* : booleans*) $\rightsquigarrow$ true
Rule or(false, B^* : booleans*) $\rightsquigarrow$ or(B^*)

Funcon exclusive-or(_ : booleans, _ : booleans) : $\Rightarrow$ booleans
Alias xor = exclusive-or

exclusive-or(B_1, B_2) is exclusive disjunction.

Rule exclusive-or(false, false) $\rightsquigarrow$ false
Rule exclusive-or(false, true) $\rightsquigarrow$ true
Rule exclusive-or(true, false) $\rightsquigarrow$ true
Rule exclusive-or(true, true) $\rightsquigarrow$ false