Maybe

fmap id = mapMaybe id

mapMaybe id Nothing = Nothing = id Nothing

mapMaybe id (Just a) = Just (id a) = Just a = id (Just a)

∴ fmap id = id

fmap (f· g) Nothing = mapMaybe (f·g) Nothing = Nothing

(fmap f · fmap g) Nothing = fmap f (fmap g Nothing)
= fmap f (mapMaybe g Nothing)
= fmap f Nothing
= Nothing

fmap (f·g) (Just a) = mapMaybe (f·g) (Just a) = Just (f·g a)

(fmap f · fmap g) (Just a) = fmapf (fmap g (Just a))
= fmap f (Just (~~f~~ g a))
= mapMaybe f (Just (g a))
= Just (f(g a))
= Just (f · g a)

∴ fmap (f·g) = fmap f · fmap g

∴ Maybe 满足 Functor Law

fmap id = map id

~~map id [ ] = [ ] = id [ ]~~

~~map id (a:as) = id a : map id as~~

引理 map id xs = xs

对 xs 递推

递推基 xs = [ ]

map id [ ] = [ ] （定义）

递推步骤 xs = x:xs'

map id xs = map id (x:xs')
= id x : map id xs' （定义）
= x : map id xs' （id 定义）
= x:xs' （递推假设）
= xs

∴ map id xs = xs = id xs

命题 map (f·g) xs = (map f · map g) xs

对 xs 递推

递推基 xs = [ ]

map (f·g) [ ] = [ ] 定义

(map f · map g) ~~xs~~ [ ] = (map f) (map g [ ]) = map f [ ] = [ ]

∴ map (f·g) [ ] = (map f · map g) [ ]

递推步骤 $xs = x : xs'$

$$
\begin{aligned}
map\ (f \cdot g)\ xs &= map\ (f \cdot g)\ (x : xs') \\
&= (f \cdot g)\ x : map\ (f \cdot g)\ xs' \quad \text{(定义)} \\
&= f \cdot g\ x : (map\ f \cdot map\ g)\ xs' \quad \text{(递推假设)}
\end{aligned}
$$

$$
\begin{aligned}
(map\ f \cdot map\ g)\ xs &= (map\ f \cdot map\ g)\ (x : xs') \\
&= (map\ f)\ (map\ g\ x : xs') \\
&= (map\ f)\ (g\ x : map\ g\ xs') \quad \text{(定义)} \\
&= f\ g\ x : map\ f\ (map\ g\ xs') \quad \text{(定义)} \\
&= f\ g\ x : (map\ f \cdot map\ g)\ xs' \quad \text{(. 定义)}
\end{aligned}
$$

$\therefore map\ (f \cdot g)\ xs = (map\ f \cdot map\ g)\ xs$

$\therefore$ [ ] 满足 Functor Law

Either

$fmap\ id = mapEither\ id$

$mapEither\ id\ (Left\ c) = Left\ c = id\ (Left\ c)$

$mapEither\ id\ (Right\ a) = Right\ (id\ a) = Right\ a = id\ (Right\ a)$

$\therefore fmap\ id = id$

$fmap\ (f \cdot g)\ (Left\ c) = mapEither\ (f \cdot g)\ (Left\ c) = Left\ c$

$$
\begin{aligned}
(fmap\ f \cdot fmap\ g)\ (Left\ c) &= fmap\ f\ (fmap\ g\ Left\ c) = fmap\ f\ (Left\ c) \\
&= Left\ c
\end{aligned}
$$

$fmap\ (f \cdot g)\ (Right\ a) = mapEither\ (f \cdot g)\ (Right\ a) = Right\ (f \cdot g\ a)$

$$
\begin{aligned}
(fmap\ f \cdot fmap\ g)\ (Right\ a) &= fmap\ f\ (fmap\ g\ Right\ a) \\
&= fmap\ f\ (mapEither\ g\ Right\ a) \\
&= fmap\ f\ Right\ (g\ a) \\
&= Right\ (f\ (g\ a)) \\
&= Right\ (f \cdot g\ a)
\end{aligned}
$$

$\therefore fmap\ (f \cdot g) = fmap\ f \cdot fmap\ g$

$\therefore$ Either 满足 Functor Law