0.0.1 At Key

The operation $atKey$ will return the Value v at some specified Key k.

$$
\begin{array}{|l}
\hline
AtKey[KV, K] \\
m? : KV \\
v! : V \\
k? : K \\
atKey\_ : KV \times K \pfun V \\
\hline
v! = atKey(m?\,, k?\,) \bullet \\
\qquad let\ coll == ((\operatorname{seq} m?\,) \rhd (k?\,, m?_{k?}\,)) \Rightarrow \langle (k?\,, m?_{k?}\,) \rangle \iff k? \in \operatorname{dom} m? \\
\quad = (second(head(coll)) \iff k? \mapsto m?_{k?} \in coll) \lor \\
\qquad (\emptyset \iff k? \notin \operatorname{dom} m?\,) \\
\hline
\end{array}
$$

In the schema above, $coll$ is the result of filtering for $(k?\,, m?_{k?}\,)$ within $\operatorname{seq} m?$. If the mapping was in the original $m?$, it will also be in the sequence of mappings. This means we can filter over the sequence to look for the mapping and if found, it is returned as $\langle (k?\,, m?_{k?}\,) \rangle$. To return the mapping itself, $head(coll)$ is used to extract the mapping such that the value mapped to $k?$ can be returned.

$$v! = atKey(m?\,, k?\,) = second(head(coll)) = m?_{k?} \bullet m?_{k?} : V \iff k? \in \operatorname{dom} m?$$

The following examples demonstrate the properties of $atKey$

$$
\begin{array}{ll}
M = \langle\!\langle k_0 v_{k_0}, k_1 v_{k_1} \rangle\!\rangle & \\
\quad k_0 = abc \ \land v_{k_0} = 123 & [k_0 v_{k_0} = abc \mapsto 123] \\
\quad k_1 = def \ \land v_{k_1} = xyz \mapsto 456 & [k_1 v_{k_1} = def \mapsto xyz \mapsto 456] \\
atKey(M, abc) = 123 & \\
atKey(M, def) = xyz \mapsto 456 & \\
atKey(M, foo) = \emptyset &
\end{array}
$$