$$
\begin{array}{rlll}
\textsf{BinOp} & \oplus & ::= & \texttt{Product} \mid \texttt{Sum} \mid \texttt{Arrow} \\
\textsf{Kind} & \kappa & ::= & \texttt{Ty} \mid \texttt{KHole} \mid \texttt{S}(\tau) \\
\textsf{ConstantTypes} & c & ::= & \texttt{Int} \mid \texttt{Float} \mid \texttt{Bool} \\
\textsf{UserHTyp} & \hat{\tau} & ::= & c \mid \hat{\tau}_1 \oplus \hat{\tau}_2 \mid \texttt{list}(\hat{\tau}) \mid \llparenthesis\rrparenthesis^u \mid \llparenthesis\hat{\tau}\rrparenthesis^u \\
\textsf{InternalHTyp} & \tau & ::= & c \mid \tau_1 \oplus \tau_2 \mid \texttt{list}(\tau) \mid \llparenthesis\rrparenthesis^u \mid \llparenthesis\tau\rrparenthesis^u \\
\textsf{TypeVars} & t & & \\
\textsf{UserTypePattern} & \hat{\rho} & ::= & t \mid \llparenthesis\rrparenthesis^u \mid \llparenthesis t\rrparenthesis^u \\
\textsf{UserExpression} & e & ::= & \texttt{type}\ \hat{\rho}\ \texttt{=}\ \hat{\tau}\ \texttt{in}\ e \mid \textit{elided} \\
\textsf{InternalExpression} & \tau & ::= & \texttt{type}\ \hat{\rho}\ \texttt{=}\ \tau\ \texttt{in}\ d \mid \textit{elided}
\end{array}
$$

$\boxed{\Delta; \Phi \vdash \kappa_1 \lesssim \kappa_2}$ κ_1 is a consistent subkind of κ_2

$$
\texttt{KCHoleL}\ \frac{}{\Delta; \Phi \vdash \texttt{KHole} \lesssim \kappa}
\qquad
\texttt{KCHoleR}\ \frac{}{\Delta; \Phi \vdash \kappa \lesssim \texttt{KHole}}
\qquad
\texttt{KCRespectEquiv}\ \frac{\Delta; \Phi \vdash \kappa_1 \equiv \kappa_2}{\Delta; \Phi \vdash \kappa_1 \lesssim \kappa_2}
$$

$$
\texttt{KCSubsumption}\ \frac{\Delta; \Phi \vdash \tau : \texttt{Ty}}{\Delta; \Phi \vdash \texttt{S}(\tau) \lesssim \texttt{Ty}}
$$

$\boxed{t\ \texttt{valid}}$ t is a valid type variable

t is valid if it is not a builtin-type or keyword, begins with an alpha char or underscore, and only contains alphanumeric characters, underscores, and primes.

$\boxed{\Delta; \Phi \vdash \kappa\ \texttt{kind}}$ κ forms a kind

$$
\texttt{KFTy}\ \frac{}{\Delta; \Phi \vdash \texttt{Ty}\ \texttt{kind}}
\qquad
\texttt{KFHole}\ \frac{}{\Delta; \Phi \vdash \texttt{KHole}\ \texttt{kind}}
\qquad
\texttt{KFSing}\ \frac{\Delta; \Phi \vdash \tau : \texttt{Ty}}{\Delta; \Phi \vdash \texttt{S}(\tau)\ \texttt{kind}}
$$

$\boxed{\Delta; \Phi \vdash \kappa_1 \equiv \kappa_2}$ κ_1 is equivalent to κ_2

$$\texttt{KERefl}\quad \frac{}{\Delta; \Phi \vdash \kappa \equiv \kappa} \qquad \texttt{KESymm}\quad \frac{\Delta; \Phi \vdash \kappa_1 \equiv \kappa_2}{\Delta; \Phi \vdash \kappa_2 \equiv \kappa_1} \qquad \texttt{KETrans}\quad \frac{\Delta; \Phi \vdash \kappa_1 \equiv \kappa_2 \qquad \Delta; \Phi \vdash \kappa_2 \equiv \kappa_3}{\Delta; \Phi \vdash \kappa_1 \equiv \kappa_3}$$

$$\texttt{KESingEquiv}\quad \frac{\Delta; \Phi \vdash \tau_1 \equiv \tau_2 : \texttt{Ty}}{\Delta; \Phi \vdash \texttt{S}(\tau_1) \equiv \texttt{S}(\tau_2)}$$

$\boxed{\Delta; \Phi \vdash \tau : \kappa}$ τ is assigned non-singleton kind κ

$$\texttt{KAConst}\quad \frac{}{\Delta; \Phi \vdash c : \texttt{Ty}} \qquad \texttt{KAVar}\quad \frac{t : \kappa_1 \in \Phi \qquad \Delta; \Phi \vdash \kappa_1 \Rrightarrow \lceil \kappa_2 \rceil}{\Delta; \Phi \vdash t : \kappa_2}$$

$$\texttt{KABinOp}\quad \frac{\Delta; \Phi \vdash \tau_1 : \texttt{Ty} \qquad \Delta; \Phi \vdash \tau_2 : \texttt{Ty}}{\Delta; \Phi \vdash \tau_1 \oplus \tau_2 : \texttt{Ty}} \qquad \texttt{KAList}\quad \frac{\Delta; \Phi \vdash \tau : \texttt{Ty}}{\Delta; \Phi \vdash \texttt{list}(\tau) : \texttt{Ty}}$$

$$\texttt{KAEHole}\quad \frac{u :: \kappa[\Phi'] \in \Delta \qquad \Delta; \Phi \vdash \sigma : \Phi'}{\Delta; \Phi \vdash (\!|\,|\!)^u_\sigma : \kappa}$$

$$\texttt{KANEHole}\quad \frac{\Delta; \Phi \vdash \tau : \kappa' \qquad u :: \kappa[\Phi'] \in \Delta \qquad \Delta; \Phi \vdash \sigma : \Phi'}{\Delta; \Phi \vdash (\!|\tau|\!)^u_\sigma : \kappa}$$

$\boxed{\Delta; \Phi \vdash \tau : \kappa_1 \Rrightarrow \lfloor \kappa_2 \rfloor}$ τ of kind κ_1 is self-recognized to consistent subkind κ_2

$$\texttt{KSRTy}\quad \frac{\Delta; \Phi \vdash \tau : \texttt{Ty}}{\Delta; \Phi \vdash \tau : \texttt{Ty} \Rrightarrow \lfloor \texttt{S}(\tau) \rfloor} \qquad \texttt{KSRHole}\quad \frac{\Delta; \Phi \vdash \tau : \texttt{KHole}}{\Delta; \Phi \vdash \tau : \texttt{KHole} \Rrightarrow \lfloor \texttt{KHole} \rfloor}$$

$\boxed{\Delta; \Phi \vdash \kappa_1 \Rrightarrow \lceil \kappa_2 \rceil}$ κ_1 is unrecognized to consistent superkind κ_2

$$\texttt{KUSing}\quad \frac{\Delta; \Phi \vdash \tau : \texttt{Ty} \qquad \Delta; \Phi \vdash \tau : \texttt{Ty} \Rrightarrow \lfloor \texttt{S}(\tau) \rfloor}{\Delta; \Phi \vdash \texttt{S}(\tau) \Rrightarrow \lceil \texttt{Ty} \rceil}$$

$$\texttt{KUHole}\quad \frac{}{\Delta; \Phi \vdash \texttt{KHole} \Rrightarrow \lceil \texttt{KHole} \rceil}$$

$\boxed{\Delta; \Phi \vdash \tau_1 \equiv \tau_2 : \kappa}$ τ_1 is equivalent to τ_2 and has kind κ_2

$$\texttt{KCERefl}\quad \frac{\Delta; \Phi \vdash \tau : \kappa}{\Delta; \Phi \vdash \tau \equiv \tau : \kappa}$$

$$\texttt{KCESymm}\quad \frac{\Delta; \Phi \vdash \tau_1 \equiv \tau_2 : \kappa}{\Delta; \Phi \vdash \tau_2 \equiv \tau_1 : \kappa}$$

$$\texttt{KCETrans}\quad \frac{\Delta; \Phi \vdash \tau_1 \equiv \tau_2 : \kappa \qquad \Delta; \Phi \vdash \tau_2 \equiv \tau_3 : \kappa}{\Delta; \Phi \vdash \tau_1 \equiv \tau_3 : \kappa}$$

$$\texttt{KCESingEquiv}\quad \frac{\Delta; \Phi \vdash \tau_1 : \texttt{S}(\tau_2)}{\Delta; \Phi \vdash \tau_1 \equiv \tau_2 : \texttt{Ty}}$$

$\boxed{\Phi \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta}$ $\hat{\tau}$ synthesizes kind κ and elaborates to τ

$$\frac{}{\Phi \vdash c \Rightarrow \mathtt{S}(c) \rightsquigarrow c \dashv \cdot}\ \mathtt{TElabSConst}$$

$$\frac{\begin{array}{ll}\Phi \vdash \hat{\tau}_1 \Leftarrow \mathtt{Ty} \rightsquigarrow \tau_1 : \mathtt{Ty} \dashv \Delta_1 & \Delta_1; \Phi \vdash \tau_1 : \mathtt{Ty} \Rrightarrow \lfloor \mathtt{S}(\tau_1) \rfloor \\ \Phi \vdash \hat{\tau}_2 \Leftarrow \mathtt{Ty} \rightsquigarrow \tau_2 : \mathtt{Ty} \dashv \Delta_2 & \Delta_2; \Phi \vdash \tau_2 : \mathtt{Ty} \Rrightarrow \lfloor \mathtt{S}(\tau_2) \rfloor \end{array}}{\Phi \vdash \hat{\tau}_1 \oplus \hat{\tau}_2 \Rightarrow \mathtt{S}(\tau_1 \oplus \tau_2) \rightsquigarrow \tau_1 \oplus \tau_2 \dashv \Delta_1 \cup \Delta_2}\ \mathtt{TElabSBinOp}$$

$$\frac{\Phi \vdash \hat{\tau} \Leftarrow \mathtt{Ty} \rightsquigarrow \tau : \mathtt{Ty} \dashv \Delta \qquad \Delta; \Phi \vdash \tau : \mathtt{Ty} \Rrightarrow \lfloor \mathtt{S}(\tau) \rfloor}{\Phi \vdash \mathtt{list}(\hat{\tau}) \Rightarrow \mathtt{S}(\mathtt{list}(\tau)) \rightsquigarrow \mathtt{list}(\tau) \dashv \Delta}\ \mathtt{TElabSList}$$

$$\frac{t : \kappa \in \Phi}{\Phi \vdash t \Rightarrow \kappa \rightsquigarrow t \dashv \cdot}\ \mathtt{TElabSVar} \qquad \frac{t \notin \Phi}{\Phi \vdash t \Rightarrow \mathtt{KHole} \rightsquigarrow (\!|t|\!)^u_{\mathsf{id}(\Phi)} \dashv u :: (\!|\,|\!)[\Phi]}\ \mathtt{TElabSUVar}$$

$$\frac{}{\Phi \vdash (\!|\,|\!)^u \Rightarrow \mathtt{KHole} \rightsquigarrow (\!|\,|\!)^u_{\mathsf{id}(\Phi)} \dashv u :: (\!|\,|\!)[\Phi]}\ \mathtt{TElabSHole}$$

$$\frac{\Phi \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta}{\Phi \vdash (\!|\hat{\tau}|\!)^u \Rightarrow \mathtt{KHole} \rightsquigarrow (\!|\tau|\!)^u_{\mathsf{id}(\Phi)} \dashv \Delta, u :: (\!|\,|\!)[\Phi]}\ \mathtt{TElabSNEHole}$$

$\boxed{\Phi \vdash \hat{\tau} \Leftarrow \kappa_1 \rightsquigarrow \tau : \kappa_2 \dashv \Delta}$ $\hat{\tau}$ analyzes against kind κ_1 and elaborates to τ of consistent subkind κ_2

TElabASubsume

$$\frac{\hat{\tau} \neq (\!|\!)^u \qquad \hat{\tau} \neq (\!|\hat{\tau}'|\!)^u \qquad \begin{array}{c}\hat{\tau} \neq t \text{ where } t \notin \Phi \\ \Phi \vdash \hat{\tau} \Rightarrow \kappa' \rightsquigarrow \tau \dashv \Delta\end{array} \qquad \Delta; \Phi \vdash \kappa' \lesssim \kappa}{\Phi \vdash \hat{\tau} \Leftarrow \kappa \rightsquigarrow \tau : \kappa' \dashv \Delta}$$

TElabAUVar

$$\frac{t \notin \Phi}{\Phi \vdash t \Leftarrow \texttt{KHole} \rightsquigarrow (\!|t|\!)^u_{\mathsf{id}(\Phi)} : \texttt{KHole} \dashv u :: (\!|\!)[\Phi]}$$

TElabAEHole

$$\frac{}{\Phi \vdash (\!|\!)^u \Leftarrow \kappa \rightsquigarrow (\!|\!)^u_{\mathsf{id}(\Phi)} : \kappa \dashv u :: \kappa[\Phi]}$$

TElabANEHole

$$\frac{\Phi \vdash \hat{\tau} \Rightarrow \kappa' \rightsquigarrow \tau \dashv \Delta}{\Phi \vdash (\!|\hat{\tau}|\!)^u \Leftarrow \kappa \rightsquigarrow (\!|\tau|\!)^u_{\mathsf{id}(\Phi)} : \kappa \dashv \Delta, u :: \kappa[\Phi]}$$

$\boxed{\Delta_1; \Phi_1 \vdash \tau \rhd \hat{\rho} \dashv \Phi_2; \Delta_2}$ $\hat{\rho}$ analyzes against τ yielding new tyvar and hole bindings

RESVar

$$\frac{t \ \texttt{valid} \qquad \Delta; \Phi \vdash \tau : \kappa_1 \qquad \Delta; \Phi \vdash \tau : \kappa_1 \Rrightarrow \lfloor \kappa_2 \rfloor}{\Delta; \Phi \vdash \tau \rhd t \dashv t :: \kappa_2; \cdot}$$

RESEHole

$$\frac{}{\Delta; \Phi \vdash \tau \rhd (\!|\!)^u \dashv \cdot; u :: (\!|\!)[\Phi]}$$

RESVarHole

$$\frac{\neg(t \ \texttt{valid})}{\Delta; \Phi \vdash \tau \rhd (\!|t|\!)^u \dashv \cdot; u :: (\!|\!)[\Phi]}$$

$\boxed{\Gamma; \Phi \vdash e \Rightarrow \hat{\tau} \rightsquigarrow d \dashv \Delta}$ e synthesizes type τ and elaborates to d

ESDefine

$$\frac{\begin{array}{c}\Phi_1 \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta_1 \\ \Delta_1; \Phi_1 \vdash \tau \rhd \hat{\rho} \dashv \Phi_2; \Delta_2 \qquad \Gamma; \Phi_1 \cup \Phi_2 \vdash e \Rightarrow \tau_1 \rightsquigarrow d \dashv \Delta_3\end{array}}{\Gamma; \Phi_1 \vdash \texttt{type}\ \hat{\rho}\ \texttt{=}\ \hat{\tau}\ \texttt{in}\ e \Rightarrow \tau_1 \rightsquigarrow \texttt{type}\ \hat{\rho}\ \texttt{=}\ \tau\ \texttt{in}\ d \dashv \Delta_1 \cup \Delta_2 \cup \Delta_3}$$

$\boxed{\Delta; \Gamma; \Phi \vdash d : \tau}$ d is assigned type τ

$$\texttt{DEDefine}\quad \frac{\Delta; \Phi_1 \vdash \tau \rhd \hat{\rho} \dashv \Phi_2; \Delta \qquad \Delta; \Gamma; \Phi_1 \cup \Phi_2 \vdash d : \tau}{\Delta; \Gamma; \Phi_1 \vdash \texttt{type}\ \hat{\rho}\ \texttt{=}\ \tau\ \texttt{in}\ d : \tau}$$

Theorem 1 (Well-Kinded Elaboration)
(1) If $\Phi \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta$ *and* $\Delta; \Phi \vdash \tau : \kappa'$ *then* $\Delta; \Phi \vdash \tau : \kappa' \Rrightarrow \lfloor \kappa \rfloor$
(2) If $\Phi \vdash \hat{\tau} \Leftarrow \kappa_1 \rightsquigarrow \tau : \kappa_2 \dashv \Delta$ *then* $\Delta; \Phi \vdash \kappa_1 \lesssim \kappa_2$ *and* $\exists \kappa_3$ *such that* $\Delta; \Phi \vdash \kappa_3 \lesssim \kappa_2$ *and* $\Delta; \Phi \vdash \tau : \kappa_3$

This is like the Typed Elaboration theorem in the POPL19 paper except that (1) synthesis produces the most refined kind while assignment prefers Ty (see *Kind Assignment Ty Affinity* theorem) and (2) analysis doesn't necessarily produce the most consistent subkind for τ.

Theorem 2 (Type Elaboration Synthesis Singleton Affinity)
If $\Phi \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta$ *then* $\Delta; \Phi \vdash \tau : \kappa \Rrightarrow \lfloor \kappa \rfloor$

Type Elaboration synthesis always synthesizes the most self-recognized kind.

Theorem 3 (Kind Assignment Unicity)
If $\Delta; \Phi \vdash \tau : \kappa$ *and* $\Delta; \Phi \vdash \tau : \kappa'$ *then* $\kappa = \kappa'$

This is like the Type Unicity theorem in the POPL19 paper.

Theorem 4 (Kind Assignment Ty Affinity)
If $\Delta; \Phi \vdash \tau : \kappa$ *then* $\Delta; \Phi \vdash \tau \Rrightarrow \lceil \kappa \rceil\, \kappa$

Kind assignment assigns `Ty` rather than `S(`τ`)`. The kind is explicitly refined where needed in other places using the recognition judgment: $\Delta; \Phi \vdash \tau : \kappa_1 \Rrightarrow \lfloor \kappa_2 \rfloor$.

Theorem 5 (Type Elaboration Unicity)
(1) If $\Phi \vdash \hat{\tau} \Rightarrow \kappa_1 \rightsquigarrow \tau_1 \dashv \Delta_1$ *and* $\Phi \vdash \hat{\tau} \Rightarrow \kappa_2 \rightsquigarrow \tau_2 \dashv \Delta_2$ *then* $\kappa_1 = \kappa_2$, $\tau_1 = \tau_2$, $\Delta_1 = \Delta_2$
(2) If $\Phi \vdash \hat{\tau} \Leftarrow \kappa \rightsquigarrow \tau_1 : \kappa_1 \dashv \Delta_1$ *and* $\Phi \vdash \hat{\tau} \Leftarrow \kappa \rightsquigarrow \tau_2 : \kappa_2 \dashv \Delta_2$ *then* $\tau_1 = \tau_2$, $\kappa_1 = \kappa_2$, $\Delta_1 = \Delta_2$

This is like the Elaboration Unicity theorem in the POPL19 paper.

Theorem 6 (Kind Precision)
If $\Phi \vdash \hat{\tau} \Rightarrow \kappa \rightsquigarrow \tau \dashv \Delta$ *and* $\Phi \vdash \hat{\tau} \Leftarrow \kappa' \rightsquigarrow \tau : \kappa'' \dashv \Delta$ *then* $\Delta; \Phi \vdash \kappa \lesssim \kappa'$

Kind Precision says that any kind κ that successfully analyzes against $\hat{\tau}$ is a consistent superkind of the one that is synthesized from $\hat{\tau}$. In other words, the kind that is synthesized is the most precise.

Theorem 7 (Reognition Inversion)
If $\Delta; \Phi \vdash \tau : \kappa_2 \Rrightarrow \lfloor \kappa_1 \rfloor$ *then* $\Delta; \Phi \vdash \kappa_1 \Rrightarrow \lceil \kappa_2 \rceil$

Unrecognizing inverts self-recognition.