module *Spezifikations-Schema*

constants $Const_1, \ldots, Const_N$
variables $Var_1, \ldots, Var_N$

---

extends $Mod_1, \ldots, Mod_N$
$M \;\triangleq\;$ **instance** Mod **with**
$\quad ModConst \leftarrow Const, ModVar \leftarrow Var$

Definition von Teilformeln, inbesondere für Ausgangszustand und Actions

---

die eigentliche Spezifikation:

$$Spec \;\triangleq\; \begin{array}{l} \land\ Init \\ \land\ \Box[SomeAction]_{Vars} \end{array}$$

module *Deklarationen*

constants $Const_1, \dots, Const_N$
variables $Var_1, \dots, Var_N$

```
\begin{module}{Deklarationen}
  \begin{nodecls}
    \CONSTANTS Const_1, \dots, Const_N \\
    \VARIABLES Var_1, \dots, Var_N
  \end{nodecls}
\end{module}
```

module *Einbindung anderer Module*

$$\textbf{extends}\ Mod_1, \ldots, Mod_N$$

$$\textbf{instance}\ Mod\ \textbf{with}$$

$$\quad ModConst \leftarrow Const, ModVar \leftarrow Var$$

$$M \triangleq \textbf{instance}\ Mod\ \textbf{with}$$

$$\quad ModConst \leftarrow Const, ModVar \leftarrow Var$$

```
\begin{module}{Einbindung anderer Module}
  \begin{nodecls}
    \EXTENDS Mod_1, \dots, Mod_N \\
    \INSTANCE Mod \WITH \\
    ___ModConst <- Const, ModVar <- Var \\
    M == \INSTANCE Mod \WITH \\
    ___ModConst <- Const, ModVar <- Var
  \end{nodecls}
\end{module}
```

module *Ausgangszustand*

$$
\begin{aligned}
Init \;\triangleq\; & \land\ Const_1 = Wert_1 \\
& \land\ \ldots \\
& \land\ Const_N = Wert_N \\
& \land\ Var_1 = InitWert_1 \\
& \land\ \ldots \\
& \land\ Var_N = InitWert_N
\end{aligned}
$$

```
\begin{module}{Ausgangszustand}
  \begin{nodecls}
    Init == \begin{conj}
              Const_1 = Wert_1 \\
              \dots \\
              Const_N = Wert_N \\
              Var_1 = InitWert_1 \\
              \dots \\
              Var_N = InitWert_N \\
            \end{conj} \\
  \end{nodecls}
\end{module}
```

module *Actions*

$$
\begin{aligned}
Act_1 \;\triangleq\;& \land\; PreConditions_1\\
& \land\; PostConditions_1\\
& \land\; \textsf{unchanged}\;\; UnchangedVars_1\\
\ldots&\\
Act_N \;\triangleq\;& \land\; PreConditions_N\\
& \land\; PostConditions_N\\
& \land\; \textsf{unchanged}\;\; UnchangedVars_N
\end{aligned}
$$

```
\begin{module}{Actions}
  \begin{nodecls}
    Act_1 == \begin{conj}
                 PreConditions_1 \\
                 PostConditions_1 \\
                 \UNCHANGED UnchangedVars_1 \\
               \end{conj} \\
    \dots \\
    Act_N == \begin{conj}
                 PreConditions_N \\
                 PostConditions_N \\
                 \UNCHANGED UnchangedVars_N \\
               \end{conj} \\
  \end{nodecls}
\end{module}
```

module *Die Gesamt-Spezifikation*

$$
\begin{aligned}
Vars &\triangleq \langle Var_1, \ldots, Var_N \rangle \\
SomeAction &\triangleq \lor\ Act_1 \\
&\quad\ \lor\ \ldots \\
&\quad\ \lor\ Act_N \\
Spec &\triangleq \land\ Init \\
&\quad\ \land\ \Box[SomeAction]_{Vars}
\end{aligned}
$$

```
\begin{module}{Die Gesamt-Spezifikation}
  \begin{nodecls}
    Vars == << Var_1, \dots, Var_N >> \\
    SomeAction == \begin{disj}
                    Act_1 \\
                    \dots \\
                    Act_N \\
                  \end{disj} \\
    Spec == \begin{conj}
              Init \\
              [][SomeAction]_{Vars} \\
            \end{conj}
  \end{nodecls}
\end{module}
```

Das Φ-Beispiel aus der Vorlesung:

$$
\begin{aligned}
Init_\Phi &\triangleq (x = 0) \land (y = 0)\\
\mathcal{M}_1 &\triangleq (x' = x + 1) \land (y' = y)\\
\mathcal{M}_2 &\triangleq (y' = y + 1) \land (x' = x)\\
\mathcal{M} &\triangleq \mathcal{M}_1 \lor \mathcal{M}_2\\
\Phi &\triangleq Init_\Phi \land \Box[\mathcal{M}]_{\langle x,y \rangle}
\end{aligned}
$$

module Φ-*Beispiel*

variables x, y

$$
\begin{aligned}
Init_\Phi &\triangleq \land\ x = 0\\
&\ \land\ y = 0\\
\mathcal{M}_1 &\triangleq \land\ x' = x + 1\\
&\ \land\ \text{unchanged } y\\
\mathcal{M}_2 &\triangleq \land\ y' = y + 1\\
&\ \land\ \text{unchanged } x\\
\mathcal{M} &\triangleq \lor\ \mathcal{M}_1\\
&\ \lor\ \mathcal{M}_2\\
\Phi &\triangleq \land\ Init_\Phi\\
&\ \land\ \Box[\mathcal{M}]_{\langle x,y \rangle}
\end{aligned}
$$

```
\begin{module}{$\Phi$-Beispiel}
  \begin{nodecls}
    \VARIABLES x, y
  \end{nodecls}

  \begin{nodecls}
    Init_{\Phi} == \begin{conj}
                     x = 0 \\
                     y = 0 \\
                   \end{conj} \\
    {\cal M}_1 == \begin{conj}
                    x' = x + 1 \\
                    \UNCHANGED y \\
                  \end{conj} \\
    {\cal M}_2 == \begin{conj}
                    y' = y + 1 \\
                    \UNCHANGED x \\
                  \end{conj} \\
    {\cal M} == \begin{disj}
                  {\cal M}_1 \\
                  {\cal M}_2 \\
                \end{disj} \\
    \Phi == \begin{conj}
              Init_{\Phi} \\
              [][{\cal M}]_{<< x, y >>} \\
            \end{conj}
  \end{nodecls}
\end{module}
```

module *Counter*

variable v

$$
\begin{aligned}
Init &\triangleq v = 0 \\
Step &\triangleq v' = v + 1 \\
Spec &\triangleq \land\ Init \\
&\ \land\ \Box[Step]_v
\end{aligned}
$$

```
\begin{module}{Counter}
  \begin{nodecls}
    \VARIABLE v
  \end{nodecls}

  \begin{nodecls}
    Init == v = 0 \\
    Step == v' = v + 1 \\
    Spec == \begin{conj}
              Init \\
              [][Step]_v \\
            \end{conj}
  \end{nodecls}
\end{module}
```

module Φ-*Beispiel-2*

variables x, y

$$C_1 \triangleq \textbf{instance } Counter \textbf{ with } v \leftarrow x$$
$$C_2 \triangleq \textbf{instance } Counter \textbf{ with } v \leftarrow y$$

$$Init_\Phi \triangleq \begin{array}{l} \land\ C_1.Init \\ \land\ C_2.Init \end{array}$$

$$\mathcal{M}_1 \triangleq \begin{array}{l} \land\ C_1.Step \\ \land\ \text{unchanged } y \end{array}$$

$$\mathcal{M}_2 \triangleq \begin{array}{l} \land\ C_2.Step \\ \land\ \text{unchanged } x \end{array}$$

$$\mathcal{M} \triangleq \begin{array}{l} \lor\ \mathcal{M}_1 \\ \lor\ \mathcal{M}_2 \end{array}$$

$$\Phi \triangleq \begin{array}{l} \land\ Init_\Phi \\ \land\ \Box[\mathcal{M}]_{\langle x,y \rangle} \end{array}$$