```
─────────────────────── MODULE GDPR_Time ───────────────────────
EXTENDS Naturals, TimeUtils, Sequences

CONSTANTS
    DataSubjects,
    Data,
    InitialEvents

TimePoint ≜ {e.time : e ∈ InitialEvents} ∪ {e.end_time : e ∈ InitialEvents}

EventRecordTypes ≜ {"StartProcessing", "GiveConsent", "WithdrawConsent",
                              "StartContract", "EndContract"}

Event ≜ [type : EventRecordTypes, time : TimePoint, subject : DataSubjects,
                                    data : Data, end_time : TimePoint]

LegalBasis ≜ [type : {"Consent", "Contract"},
               subject : DataSubjects,
               data : Data,
               start : TimePoint,
               end : TimePoint]

Process ≜ [subject : DataSubjects,
            data : Data,
            start : TimePoint,
            end : TimePoint]



VARIABLES
    currentTime,
    eventsToProcess,
    activeProcesses,
    activeLegalBases,
    breachesInProgress


InitialTime ≜ IF InitialEvents = {} THEN
                   [year ↦ FixedEpochYear, month ↦ 1, day ↦ 1, hour ↦ 0, minute ↦ 0]
                 ELSE MinTime(InitialEvents)

EndTime ≜ IF InitialEvents = {} THEN
                [year ↦ FixedEpochYear + 50, month ↦ 12, day ↦ 31,
                                               hour ↦ 23, minute ↦ 59]
            ELSE MaxTime(InitialEvents)
```

$$
\begin{array}{lll}
\mathit{Init} & \triangleq & \land\ \mathit{currentTime} = \mathit{InitialTime} \\
& & \land\ \mathit{eventsToProcess} = \mathit{InitialEvents} \\
& & \land\ \mathit{activeProcesses}\ \ = \{\} \\
& & \land\ \mathit{activeLegalBases} = \{\} \\
& & \land\ \mathit{breachesInProgress} = \{\}
\end{array}
$$

$$
\begin{array}{l}
\mathit{StartProcessing}(e)\ \triangleq \\
\quad \land\ e.\mathit{type} = \text{“StartProcessing”} \\
\quad \land\ \mathit{eventsToProcess}' = \mathit{eventsToProcess} \setminus \{e\} \\
\quad \land\ \mathit{currentTime}' = e.\mathit{time} \\
\quad \land\ \mathit{activeProcesses}' = \mathit{activeProcesses} \cup \{[\begin{array}{rl} \mathit{subject} & \mapsto e.\mathit{subject}, \\ \mathit{data} & \mapsto e.\mathit{data}, \\ \mathit{start} & \mapsto e.\mathit{time}, \\ \mathit{end} & \mapsto \mathit{EndTime}]\} \end{array} \\
\quad \land\ \text{UNCHANGED}\ \langle \mathit{activeLegalBases},\ \mathit{breachesInProgress} \rangle
\end{array}
$$

$$
\begin{array}{l}
\mathit{GiveConsent}(e)\ \triangleq \\
\quad \land\ e.\mathit{type} = \text{“GiveConsent”} \\
\quad \land\ \mathit{eventsToProcess}' = \mathit{eventsToProcess} \setminus \{e\} \\
\quad \land\ \mathit{currentTime}' = e.\mathit{time} \\
\quad \land\ \mathit{activeLegalBases}' = \mathit{activeLegalBases} \cup \{[\begin{array}{ll} \mathit{type} & \mapsto \text{“Consent”}, \\ \mathit{subject} & \mapsto e.\mathit{subject}, \\ \mathit{data} & \mapsto e.\mathit{data}, \\ \mathit{start} & \mapsto e.\mathit{time}, \\ \mathit{end} & \mapsto \mathit{EndTime}]\} \end{array} \\
\quad \land\ \text{UNCHANGED}\ \langle \mathit{activeProcesses},\ \mathit{breachesInProgress} \rangle
\end{array}
$$

$$
\begin{array}{l}
\mathit{WithdrawConsent}(e)\ \triangleq \\
\qquad \land\ e.\mathit{type} = \text{“WithdrawConsent”} \\
\qquad \land\ \exists\, c \in \mathit{activeLegalBases} : \\
\qquad\quad c.\mathit{type} = \text{“Consent”} \land c.\mathit{subject} = e.\mathit{subject} \land c.\mathit{data} = e.\mathit{data} \\
\qquad \land\ \text{LET}\ \mathit{consentToRemove}\ \triangleq\ \text{CHOOSE}\ c \in \mathit{activeLegalBases} : \\
\qquad\qquad\qquad\qquad c.\mathit{type} = \text{“Consent”} \land c.\mathit{subject} = e.\mathit{subject} \land c.\mathit{data} = e.\mathit{data} \\
\qquad\quad \text{IN} \\
\qquad\quad \land\ \mathit{eventsToProcess}' = \mathit{eventsToProcess} \setminus \{e\} \\
\qquad\quad \land\ \mathit{currentTime}' = e.\mathit{time} \\
\qquad\quad \land\ \mathit{activeLegalBases}' = (\mathit{activeLegalBases} \setminus \{\mathit{consentToRemove}\}) \\
\qquad\qquad\qquad\qquad \cup \{[\begin{array}{ll} \mathit{type} & \mapsto \mathit{consentToRemove}.\mathit{type}, \\ \mathit{subject} & \mapsto \mathit{consentToRemove}.\mathit{subject}, \\ \mathit{data} & \mapsto \mathit{consentToRemove}.\mathit{data}, \\ \mathit{start} & \mapsto \mathit{consentToRemove}.\mathit{start}, \\ \mathit{end} & \mapsto e.\mathit{time}]\} \end{array} \\
\qquad\quad \land\ \text{UNCHANGED}\ \langle \mathit{activeProcesses},\ \mathit{breachesInProgress} \rangle
\end{array}
$$

$$
\mathit{StartContract}(e)\ \triangleq
$$

```
    $\land\ e.type = \text{“StartContract”}$
    $\land\ eventsToProcess' = eventsToProcess \setminus \{e\}$
    $\land\ currentTime' = e.time$
    $\land\ activeLegalBases' = activeLegalBases \cup \{[type \mapsto \text{“Contract”},$
                                              $subject \mapsto e.subject,$
                                                 $data \mapsto e.data,$
                                                $start \mapsto e.time,$
                                                  $end \mapsto e.end\_time]\}$
    $\land\ \text{UNCHANGED}\ \langle activeProcesses,\ breachesInProgress \rangle$

$EndContract(e) \triangleq$
    $\land\ e.type = \text{“EndContract”}$
    $\land\ \exists\, c \in activeLegalBases :$
      $c.type = \text{“Contract”} \land c.subject = e.subject \land c.data = e.data$
    $\land\ \text{LET}\ contractToEnd \triangleq \text{CHOOSE}\ c \in activeLegalBases :$
                              $c.type = \text{“Contract”} \land c.subject = e.subject \land c.data = e.data$
      $\text{IN}$
        $\land\ contractToEnd \in activeLegalBases$
        $\land\ eventsToProcess' = eventsToProcess \setminus \{e\}$
        $\land\ currentTime' = e.time$
        $\land\ activeLegalBases' = (activeLegalBases \setminus \{contractToEnd\})$
                                 $\cup \{[type \mapsto contractToEnd.type,$
                                      $subject \mapsto contractToEnd.subject,$
                                      $data \mapsto contractToEnd.data,$
                                      $start \mapsto contractToEnd.start,$
                                      $end \mapsto e.time]\}$
        $\land\ \text{UNCHANGED}\ \langle activeProcesses,\ breachesInProgress \rangle$

$IsLawful(p) \triangleq$
    $\exists\, l \in activeLegalBases :$
       $\land\ p.subject = l.subject$
       $\land\ p.data = l.data$
       $\land\ TimeBetween(l.start,\ l.end,\ currentTime)$
       $\land\ TimeBetween(p.start,\ p.end,\ currentTime)$


$BreachOccurs \triangleq$
       $\exists\, p \in activeProcesses :$
          $\land\ \neg IsLawful(p)$
          $\land\ [process \mapsto p,\ status \mapsto \text{“Pending”}] \notin breachesInProgress$
          $\land\ breachesInProgress' = breachesInProgress$
                                 $\cup \{[process \mapsto p,$
                                       $status \mapsto \text{“Pending”},$
                                   $breachTime \mapsto currentTime$
                                    $]$
                                   $\}$
          $\land \text{UNCHANGED } \langle \mathit{currentTime}, \mathit{activeProcesses}, \mathit{activeLegalBases}, \mathit{eventsToProcess} \rangle$

$\mathit{ReportBreach} \triangleq$
   $\exists\, b \in \mathit{breachesInProgress} :$
      $\land\ b.\mathit{status} = \text{“Pending”}$
      $\land\ \mathit{breachesInProgress}' = (\mathit{breachesInProgress} \setminus \{b\}) \cup \{[b \text{ EXCEPT } !.\mathit{status} = \text{“Reported”}]\}$
      $\land \text{UNCHANGED } \langle \mathit{currentTime}, \mathit{activeProcesses}, \mathit{activeLegalBases}, \mathit{eventsToProcess} \rangle$
$\mathit{Next} \triangleq$
    Event-driven actions
    $\lor\ \exists\, e \in \mathit{eventsToProcess} :$
       $\land\ e.\mathit{time} = \mathit{MinTime}(\mathit{eventsToProcess})$
       $\land\ \lor\ \mathit{GiveConsent}(e)$
          $\lor\ \mathit{WithdrawConsent}(e)$
          $\lor\ \mathit{StartProcessing}(e)$
          $\lor\ \mathit{StartContract}(e)$
          $\lor\ \mathit{EndContract}(e)$
    State-driven actions
    $\lor\ \mathit{BreachOccurs}$
    $\lor\ \mathit{ReportBreach}$
```

```
$\mathit{TypeInvariant} \triangleq$
    $\land\ \mathit{currentTime} \in \mathit{TimePoint}$
    $\land\ \mathit{eventsToProcess} \subseteq \mathit{InitialEvents}$
    $\land\ \mathit{activeProcesses}\ \ \subseteq \mathit{Process}$
    $\land\ \mathit{activeLegalBases} \subseteq \mathit{LegalBasis}$
    $\land\ \mathit{breachesInProgress} \subseteq [\mathit{breachTime} : \mathit{TimePoint}, \mathit{status} : \{\text{“Pending”}, \text{“Reported”}\}]$

 Rule 1: Legal Basis Requirement If personal data is being processed, there must be a legal basis
 for it.
$\mathit{AllProcessingIsLawful} \triangleq$
   $\forall\, p \in \mathit{activeProcesses} :$
      $\exists\, l \in \mathit{activeLegalBases} :$
         $\land\ p.\mathit{subject} = l.\mathit{subject}$
         $\land\ p.\mathit{data} = l.\mathit{data}$
         $\land\ \mathit{TimeBetween}(l.\mathit{start}, l.\mathit{end}, \mathit{currentTime})$
 Rule 2: Legal Basis Types
A legal basis must be a recognized type, such as consent or contract.
$\mathit{LegalBasesHaveValidType} \triangleq$
   $\forall\, l \in \mathit{activeLegalBases} : l.\mathit{type} \in \{\text{“Consent”}, \text{“Contract”}\}$

 Rule 3: Consent Timing
Consent must be obtained before processing starts and remain valid during processing.
$\mathit{ConsentTimingIsValid} \triangleq$
   $\forall\, p \in \mathit{activeProcesses} :$
      $(\exists\, l \in \mathit{activeLegalBases} :$
        $\land\ p.subject = l.subject$
        $\land\ p.data = l.data$
        $\land\ l.type\ = \text{“Consent”}$
        $\land\ Before(l.start,\ p.start)$
    $)$
```

Rule 4: *Contract* Timing
Contract-based processing is only lawful during the contract term.

```
$ContractTimingIsValid\ \triangleq$
  $\forall\, p \in activeProcesses :$
    $(\exists\, l \in activeLegalBases :$
        $\land\ p.subject = l.subject$
        $\land\ p.data = l.data$
        $\land\ l.type\ = \text{“Contract”}$
        $\land\ TimeBetween(l.start,\ l.end,\ currentTime)$
    $)$
```

Rule 5: Breach Reporting Deadline
Guarantees that data breaches are reported within 72 hours of discovery.

```
$BreachReportedOnTime\ \triangleq$
  $\forall\, b \in breachesInProgress :$
    $(b.status = \text{“Pending”}) \Rightarrow Within72Hours(b.breachTime,\ currentTime)$
```