MODULE *GDPR_Time*

EXTENDS *Naturals*, *TimeUtils*, *Sequences*

CONSTANTS
DataSubjects,
Data,
InitialEvents

$$TimePoint \triangleq \{e.time : e \in InitialEvents\} \cup \{e.end\_time : e \in InitialEvents\}$$

$$EventRecordTypes \triangleq \{\text{"StartProcessing"}, \text{"GiveConsent"}, \text{"WithdrawConsent"}, \text{"StartContract"}, \text{"EndContract"}\}$$

$$Event \triangleq [type : EventRecordTypes, time : TimePoint, subject : DataSubjects, data : Data, end\_time : TimePoint]$$

$$LegalBasis \triangleq [type : \{\text{"Consent"}, \text{"Contract"}\}, subject : DataSubjects, data : Data, start : TimePoint, end : TimePoint]$$

$$Process \triangleq [subject : DataSubjects, data : Data, start : TimePoint, end : TimePoint]$$

VARIABLES
currentTime,
eventsToProcess,
activeProcesses,
activeLegalBases,
breachesInProgress

$$InitialTime \triangleq \text{IF } InitialEvents = \{\} \text{ THEN } [year \mapsto FixedEpochYear, month \mapsto 1, day \mapsto 1, hour \mapsto 0, minute \mapsto 0] \text{ ELSE } MinTime(InitialEvents)$$

$$EndTime \triangleq \text{IF } InitialEvents = \{\} \text{ THEN } [year \mapsto FixedEpochYear + 50, month \mapsto 12, day \mapsto 31, hour \mapsto 23, minute \mapsto 59] \text{ ELSE } MaxTime(InitialEvents)$$

$$Init \triangleq \land currentTime = InitialTime \land eventsToProcess = InitialEvents$$

$$
\begin{aligned}
&\wedge\ activeProcesses\ \ = \{\} \\
&\wedge\ activeLegalBases = \{\} \\
&\wedge\ breachesInProgress = \{\}
\end{aligned}
$$

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

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

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

$$
\begin{aligned}
&StartContract(e)\ \triangleq \\
&\quad \wedge\ e.type = \text{“StartContract”} \\
&\quad \wedge\ eventsToProcess' = eventsToProcess \setminus \{e\}
\end{aligned}
$$

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

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

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

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

$ReportBreach \triangleq$

$\exists\, b \in breachesInProgress :$
$\land\ b.status = \text{“Pending”}$
$\land\ breachesInProgress' = (breachesInProgress \setminus \{b\}) \cup \{[b \text{ EXCEPT } !.status = \text{“Reported”}]\}$
$\land$ UNCHANGED $\langle currentTime,\ activeProcesses,\ activeLegalBases,\ eventsToProcess\rangle$

$Next \triangleq$

Event-driven actions

$\lor \exists\, e \in eventsToProcess :$
$\land\ e.time = MinTime(eventsToProcess)$
$\land\ \lor\ GiveConsent(e)$
$\lor\ WithdrawConsent(e)$
$\lor\ StartProcessing(e)$
$\lor\ StartContract(e)$
$\lor\ EndContract(e)$

State-driven actions

$\lor\ BreachOccurs$
$\lor\ ReportBreach$

$vars \triangleq \langle activeProcesses,\ activeLegalBases,\ breachesInProgress,\ eventsToProcess,\ currentTime\rangle$

$Spec \triangleq Init \land \Box[Next]_{vars} \land \mathrm{WF}_{vars}(Next)$