

---

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

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

*Init*  $\triangleq \wedge currentTime = InitialTime$   
 $\wedge eventsToProcess = InitialEvents$

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

$$\begin{aligned}
\text{StartProcessing}(e) &\triangleq \\
&\wedge e.type = \text{"StartProcessing"} \\
&\wedge \text{eventsToProcess}' = \text{eventsToProcess} \setminus \{e\} \\
&\wedge \text{currentTime}' = e.time \\
&\wedge \text{activeProcesses}' = \text{activeProcesses} \cup \{[subject \mapsto e.subject, \\
&\hspace{15em} data \mapsto e.data, \\
&\hspace{15em} start \mapsto e.time, \\
&\hspace{15em} end \mapsto \text{EndTime}]\} \\
&\wedge \text{UNCHANGED} \langle \text{activeLegalBases}, \text{breachesInProgress} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{GiveConsent}(e) &\triangleq \\
&\wedge e.type = \text{"GiveConsent"} \\
&\wedge \text{eventsToProcess}' = \text{eventsToProcess} \setminus \{e\} \\
&\wedge \text{currentTime}' = e.time \\
&\wedge \text{activeLegalBases}' = \text{activeLegalBases} \cup \{[type \mapsto \text{"Consent"}, \\
&\hspace{15em} subject \mapsto e.subject, \\
&\hspace{15em} data \mapsto e.data, \\
&\hspace{15em} start \mapsto e.time, \\
&\hspace{15em} end \mapsto \text{EndTime}]\} \\
&\wedge \text{UNCHANGED} \langle \text{activeProcesses}, \text{breachesInProgress} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{WithdrawConsent}(e) &\triangleq \\
&\wedge e.type = \text{"WithdrawConsent"} \\
&\wedge \exists c \in \text{activeLegalBases} : \\
&\hspace{2em} c.type = \text{"Consent"} \wedge c.subject = e.subject \wedge c.data = e.data \\
&\wedge \text{LET } \text{consentToRemove} \triangleq \text{CHOOSE } c \in \text{activeLegalBases} : \\
&\hspace{4em} c.type = \text{"Consent"} \wedge c.subject = e.subject \wedge c.data = e.data \\
&\text{IN} \\
&\wedge \text{eventsToProcess}' = \text{eventsToProcess} \setminus \{e\} \\
&\wedge \text{currentTime}' = e.time \\
&\wedge \text{activeLegalBases}' = (\text{activeLegalBases} \setminus \{\text{consentToRemove}\}) \\
&\hspace{10em} \cup \{[type \mapsto \text{consentToRemove.type}, \\
&\hspace{15em} subject \mapsto \text{consentToRemove.subject}, \\
&\hspace{15em} data \mapsto \text{consentToRemove.data}, \\
&\hspace{15em} start \mapsto \text{consentToRemove.start}, \\
&\hspace{15em} end \mapsto e.time]\} \\
&\wedge \text{UNCHANGED} \langle \text{activeProcesses}, \text{breachesInProgress} \rangle
\end{aligned}$$

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

$$\begin{aligned}
& \wedge \text{currentTime}' = e.\text{time} \\
& \wedge \text{activeLegalBases}' = \text{activeLegalBases} \cup \{[type \mapsto \text{"Contract"}, \\
& \quad \quad \quad \text{subject} \mapsto e.\text{subject}, \\
& \quad \quad \quad \text{data} \mapsto e.\text{data}, \\
& \quad \quad \quad \text{start} \mapsto e.\text{time}, \\
& \quad \quad \quad \text{end} \mapsto e.\text{end\_time}]\} \\
& \wedge \text{UNCHANGED } \langle \text{activeProcesses}, \text{breachesInProgress} \rangle \\
\text{EndContract}(e) & \triangleq \\
& \wedge e.\text{type} = \text{"EndContract"} \\
& \wedge \exists c \in \text{activeLegalBases} : \\
& \quad c.\text{type} = \text{"Contract"} \wedge c.\text{subject} = e.\text{subject} \wedge c.\text{data} = e.\text{data} \\
& \wedge \text{LET } \text{contractToEnd} \triangleq \text{CHOOSE } c \in \text{activeLegalBases} : \\
& \quad \quad \quad c.\text{type} = \text{"Contract"} \wedge c.\text{subject} = e.\text{subject} \wedge c.\text{data} = e.\text{data} \\
& \text{IN} \\
& \quad \wedge \text{contractToEnd} \in \text{activeLegalBases} \\
& \quad \wedge \text{eventsToProcess}' = \text{eventsToProcess} \setminus \{e\} \\
& \quad \wedge \text{currentTime}' = e.\text{time} \\
& \quad \wedge \text{activeLegalBases}' = (\text{activeLegalBases} \setminus \{\text{contractToEnd}\}) \\
& \quad \quad \cup \{[type \mapsto \text{contractToEnd.type}, \\
& \quad \quad \quad \text{subject} \mapsto \text{contractToEnd.subject}, \\
& \quad \quad \quad \text{data} \mapsto \text{contractToEnd.data}, \\
& \quad \quad \quad \text{start} \mapsto \text{contractToEnd.start}, \\
& \quad \quad \quad \text{end} \mapsto e.\text{time}]\} \\
& \quad \wedge \text{UNCHANGED } \langle \text{activeProcesses}, \text{breachesInProgress} \rangle \\
\text{IsLawful}(p) & \triangleq \\
& \exists l \in \text{activeLegalBases} : \\
& \quad \wedge p.\text{subject} = l.\text{subject} \\
& \quad \wedge p.\text{data} = l.\text{data} \\
& \quad \wedge \text{TimeBetween}(l.\text{start}, l.\text{end}, \text{currentTime}) \\
\text{BreachOccurs} & \triangleq \\
& \exists p \in \text{activeProcesses} : \\
& \quad \wedge \neg \text{IsLawful}(p) \\
& \quad \wedge [process \mapsto p, status \mapsto \text{"Pending"}] \notin \text{breachesInProgress} \\
& \quad \wedge \text{breachesInProgress}' = \text{breachesInProgress} \\
& \quad \quad \cup \{[process \mapsto p, \\
& \quad \quad \quad status \mapsto \text{"Pending"}, \\
& \quad \quad \quad \text{breachTime} \mapsto \text{currentTime} \\
& \quad \quad \quad ] \\
& \quad \quad \quad \} \\
& \quad \wedge \text{UNCHANGED } \langle \text{currentTime}, \text{activeProcesses}, \text{activeLegalBases}, \text{eventsToProcess} \rangle \\
\text{ReportBreach} & \triangleq
\end{aligned}$$

$$\begin{aligned}
& \exists b \in \text{breachesInProgress} : \\
& \quad \wedge b.\text{status} = \text{"Pending"} \\
& \quad \wedge \text{breachesInProgress}' = (\text{breachesInProgress} \setminus \{b\}) \cup \{[b \text{ EXCEPT } !.\text{status} = \text{"Reported"}]\} \\
& \quad \wedge \text{UNCHANGED } \langle \text{currentTime}, \text{activeProcesses}, \text{activeLegalBases}, \text{eventsToProcess} \rangle \\
\text{Next} & \triangleq \\
& \quad \text{Event-driven actions} \\
& \quad \vee \exists e \in \text{eventsToProcess} : \\
& \quad \quad \wedge e.\text{time} = \text{MinTime}(\text{eventsToProcess}) \\
& \quad \quad \wedge \vee \text{GiveConsent}(e) \\
& \quad \quad \quad \vee \text{WithdrawConsent}(e) \\
& \quad \quad \quad \vee \text{StartProcessing}(e) \\
& \quad \quad \quad \vee \text{StartContract}(e) \\
& \quad \quad \quad \vee \text{EndContract}(e) \\
& \quad \text{State-driven actions} \\
& \quad \vee \text{BreachOccurs} \\
& \quad \vee \text{ReportBreach} \\
\text{vars} & \triangleq \langle \text{activeProcesses}, \text{activeLegalBases}, \text{breachesInProgress}, \text{eventsToProcess}, \text{currentTime} \rangle \\
\text{Spec} & \triangleq \text{Init} \wedge \Box[\text{Next}]_{\text{vars}} \wedge \text{WF}_{\text{vars}}(\text{Next})
\end{aligned}$$


---


$$\begin{aligned}
\text{TypeInvariant} & \triangleq \\
& \quad \wedge \text{currentTime} \in \text{TimePoint} \\
& \quad \wedge \text{eventsToProcess} \subseteq \text{InitialEvents} \\
& \quad \wedge \text{activeProcesses} \subseteq \text{Process} \\
& \quad \wedge \text{activeLegalBases} \subseteq \text{LegalBasis}
\end{aligned}$$

Rule 1: Legal Basis Requirement If personal data is being processed, there must be a legal basis for it.

$$\begin{aligned}
\text{AllProcessingIsLawful} & \triangleq \\
& \quad \forall p \in \text{activeProcesses} : \\
& \quad \quad \exists l \in \text{activeLegalBases} : \\
& \quad \quad \quad \wedge p.\text{subject} = l.\text{subject} \\
& \quad \quad \quad \wedge p.\text{data} = l.\text{data} \\
& \quad \quad \quad \wedge \text{TimeBetween}(l.\text{start}, l.\text{end}, \text{currentTime})
\end{aligned}$$

Rule 2: Legal Basis Types

A legal basis must be a recognized type, such as consent or contract.

$$\begin{aligned}
\text{LegalBasesHaveValidType} & \triangleq \\
& \quad \forall l \in \text{activeLegalBases} : l.\text{type} \in \{\text{"Consent"}, \text{"Contract"}\}
\end{aligned}$$

Rule 3: Breach Reporting Deadline

Guarantees that data breaches are reported within 72 hours of discovery.

$$\begin{aligned}
\text{BreachReportedOnTime} & \triangleq \\
& \quad \forall b \in \text{breachesInProgress} :
\end{aligned}$$

$(b.status = \text{"Pending"}) \Rightarrow \text{Within72Hours}(b.breachTime, currentTime)$

THEOREM  $Spec \Rightarrow \Box \text{TypeInvariant}$

THEOREM  $Spec \Rightarrow \Box \text{AllProcessingIsLawful}$

THEOREM  $Spec \Rightarrow \Box \text{LegalBasesHaveValidType}$

THEOREM  $Spec \Rightarrow \Box \text{BreachReportedOnTime}$