A modular coloured Petri net model for complex event processing

Ariane Piel

Directrice de thèse : Christine Choppy (LIPN, U. Paris 13 & CNRS)
Encadrants: Romain Kervarc et Patrice Carle (Onera, DCPS – TCS)

MeFoSyLoMa, le 4 octobre 2013

ONERA
THE FRENCH AEROSPACE LAB

Galilée
Ecole doctorale

retour sur innovation

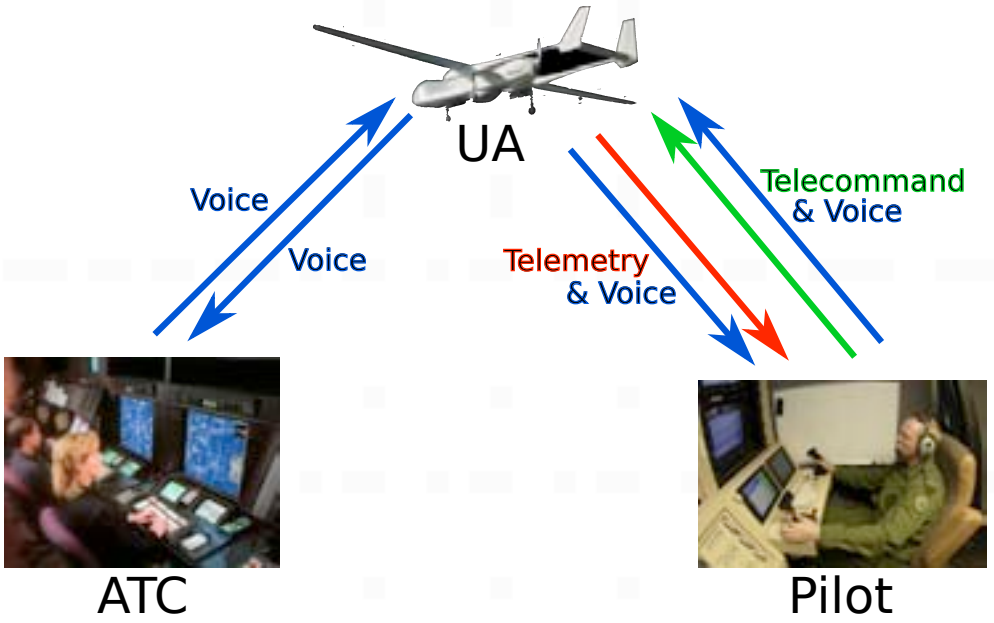
Context

- Systematic resort to simulation in the aerospace industry.
- Study of all the characteristics of complex systems before even considering of undertaking them.
- Generation of gigantic quantities of data : necessity to have an analysis assistance tool.
- ONERA has a recognised expertise in distributed simulation :
  - work on intention recognition initiated in 1998,
  - during a former Ph. D. [BER09] (2007–2009), development of an analysis and behaviour recognition tool.

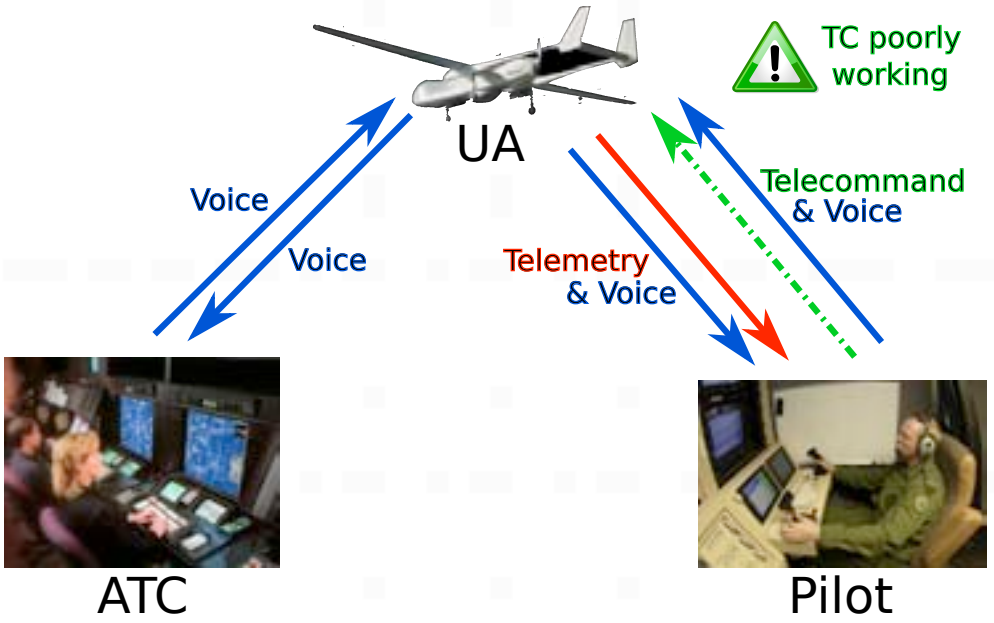
Main goals

- **Aim** : activity recognition in the framework of complex problems.
  - informatics security
  - supervision of medical environments
- Necessity of simulations since experiments may not be undertaken (because of criticality and cost for aerospace systems)
  - make use of the important quantity of data produced by the simulation in which numerous systems interact,
  - possibly, react to the data.

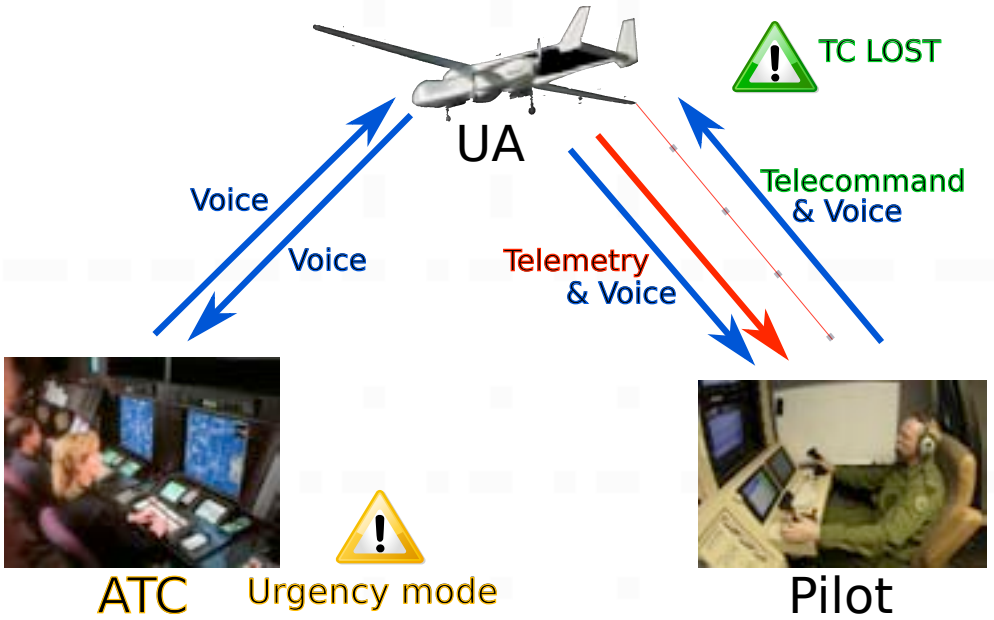
- High criticality of the system → need for strong guarantees.
- Very dynamic and highly complex data flow, be it between the agents of the system or between several systems.
- **Aim :** overseeing the consistency of the system in case of failure(s)



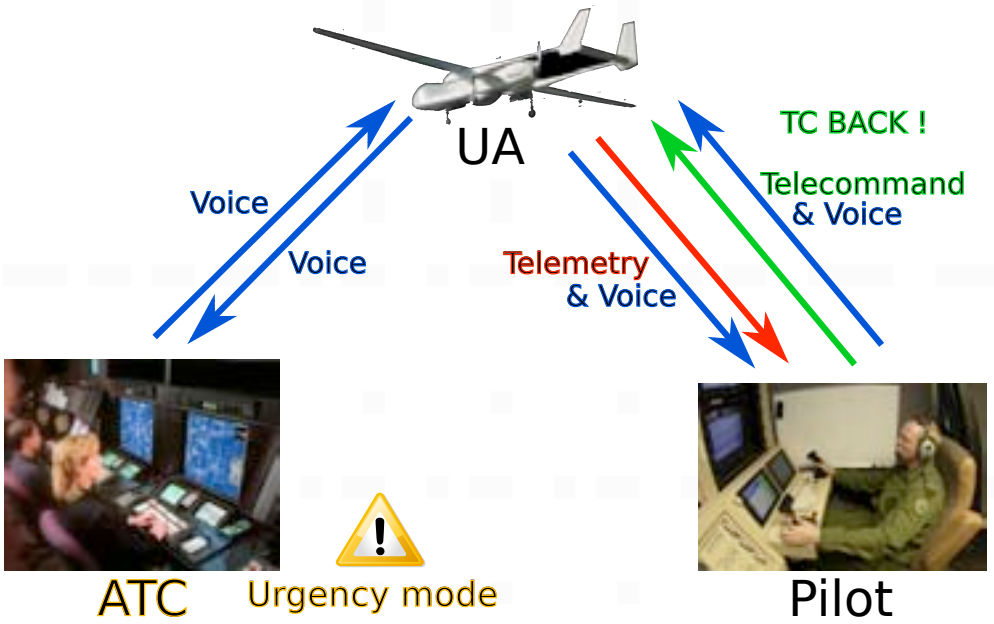
- High criticality of the system → need for strong guarantees.
- Very dynamic and highly complex data flow, be it between the agents of the system or between several systems.
- **Aim :** overseeing the consistency of the system in case of failure(s)



- High criticality of the system → need for strong guarantees.
- Very dynamic and highly complex data flow, be it between the agents of the system or between several systems.
- **Aim :** overseeing the consistency of the system in case of failure(s)



- High criticality of the system → need for strong guarantees.
- Very dynamic and highly complex data flow, be it between the agents of the system or between several systems.
- **Aim :** overseeing the consistency of the system in case of failure(s)



!! INCONSISTENT SITUATION !!

Outline

Outline

- a *single event* A
- a *disjunction* $C_1 \,||\, C_2$: at least either C_1 or C_2.
- a *conjunction* $C_1 \& C_2$: both C_1 and C_2 in any order, possibly intertwined.
- a *sequence* $C_1\ C_2$: C_1 followed by C_2.
- an *absence* $(C_1) - [C_2]$: C_1 without C_2 occurring during the recognition of C_1.

Formal definition of the chronicle language

Let $\mathfrak{N}$ be a countable set of *single event names*.

The set $X(\mathfrak{N})$ of *chronicles over* $\mathfrak{N}$ is defined inductively by these inference rules :

$$\frac{A \in \mathfrak{N}}{A \in X(\mathfrak{N})} \text{ (name)}$$

$$\frac{C_1 \in X(\mathfrak{N}) \quad C_2 \in X(\mathfrak{N})}{C_1 \,||\, C_2 \in X(\mathfrak{N})} \text{ (disjunction)} \qquad \frac{C_1 \in X(\mathfrak{N}) \quad C_2 \in X(\mathfrak{N})}{C_1 \& C_2 \in X(\mathfrak{N})} \text{ (conjunction)}$$

$$\frac{C_1 \in X(\mathfrak{N}) \quad C_2 \in X(\mathfrak{N})}{C_1\ C_2 \in X(\mathfrak{N})} \text{ (sequence)} \qquad \frac{C_1 \in X(\mathfrak{N}) \quad C_2 \in X(\mathfrak{N})}{(C_1) - [C_2] \in X(\mathfrak{N})} \text{ (absence)}$$

Chronicle recognition

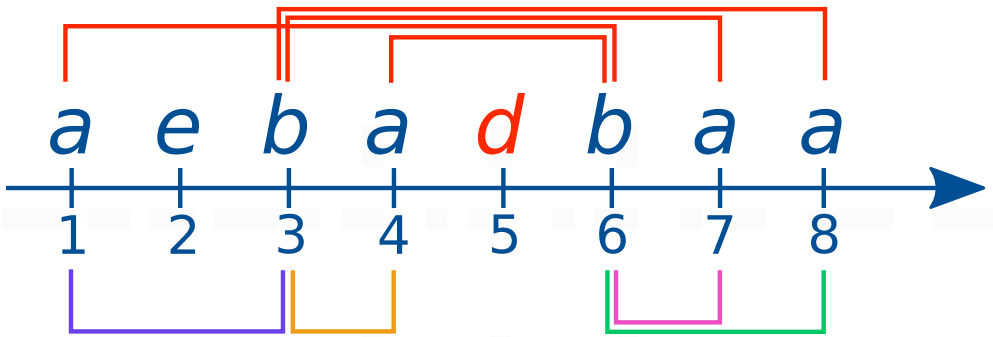
The *recognition set of C over flow φ until instant d*, denoted $R_C(\varphi, d)$, is defined by induction as follows :

- if $C = A \in \mathfrak{N}$, then
  $R_A(\varphi, d) = \{(e, t) : \exists i \ \varphi(i) = (e, t) \wedge e = a \wedge t \leq d\}$
- $R_{C_1||C_2}(\varphi, d) = \{\langle r, \perp \rangle : r \in R_{C_1}(\varphi, d)\}$
  $\cup \{\langle \perp, r \rangle : r \in R_{C_2}(\varphi, d)\}$
- $R_{C_1 \& C_2}(\varphi, d) = \{\langle r_1, r_2 \rangle : r_1 \in R_{C_1}(\varphi, d) \wedge r_2 \in R_{C_2}(\varphi, d)\}$
- $R_{C_1\ C_2}(\varphi, d) = \{\langle r_1, r_2 \rangle : r_1 \in R_{C_1}(\varphi, D) \wedge r_2 \in R_{C_2}(\varphi, d)$
  $\wedge\ T_{\max}(r_1) < T_{\min}(r_2)\}$
- $R_{(C_1)-[C_2]}(\varphi, d) = \{r_1 : r_1 \in R_{C_1}(\varphi, d) \wedge \forall r_2 \in R_{C_2}(\varphi, d)$
  $(\ T_{\min}(r_1) > T_{\min}(r_2) \vee T_{\max}(r_1) \leq T_{\max}(r_2)\ )\}$

An example : chronicle $(A\&B) - [D]$

Let a, b, d and e be events of $\mathfrak{E}$ such that $\nu(a) = A, \nu(b) = B$, $\nu(d) = D$ et $\nu(e) = E$.

Consider chronicle $C = (A\&B) - [D]$ and flow $\varphi = (a, e, b, a, d, b, a, a)$ where $\overset{\circ}{\varphi} = [\![1, 6]\!]$.



$R_C(\varphi, 8) = \{\langle 1, 3\rangle, \langle 4, 3\rangle, \langle 7, 6\rangle, \langle 8, 6\rangle\}$

A recognition model

A coloured Petri net recognition model built by induction : for any given chronicle C, a Petri net $N(C)$ computing the recognitions of C

- → modularity constraint
- → determinism, in the sense that the nets always have to provide, given an event flow, the same recognitions
- → maintaining concurrency

Outline

Why coloured Petri nets ?

- a concurrent framework
- a modular framework
  - to map the structure of the language
- a tool to visualise executions and check properties

CPN definition

A non-hierarchical *CPN* is a tuple $(P, T, A, \mathcal{B}, V, C, G, EX, I)$:

1. P is a finite set of *places*.
2. T is a finite set of *transitions* such that $P \cap T = \emptyset$.
3. $A \subseteq P \times T \uplus T \times P$ is a set of directed *arcs*.
4. $\mathcal{B}$ is a finite set of non empty *colour sets* (types).
5. $\mathcal{V}$ is a finite set of typed *variables* such that : $\forall v \in \mathcal{V} \quad Type[v] \in \mathcal{B}$.
6. $C : P \rightarrow \mathcal{B}$ is a *colour set function* assigning a colour set to places.
7. $G : T \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is a *guard function*.
8. $EX : A \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is an *arc expression function*.
9. $I : A \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is an *initialisation function*.

K. Jensen and L. M. Kristensen, "Coloured Petri nets : modelling and validation of concurrent systems", Springer, 2009.

CPN definition

A non-hierarchical *CPN* is a tuple $(P, T, A, \mathcal{B}, V, C, G, EX, I, B)$:

1. P is a finite set of *places*.
2. T is a finite set of *transitions* such that $P \cap T = \emptyset$.
3. $A \subseteq P \times T \uplus T \times P$ is a set of directed *arcs*.
4. $\mathcal{B}$ is a finite set of non empty *colour sets* (types).
5. $\mathcal{V}$ is a finite set of typed *variables* such that : $\forall v \in \mathcal{V} \quad Type[v] \in \mathcal{B}$.
6. $C : P \rightarrow \mathcal{B}$ is a *colour set function* assigning a colour set to places.
7. $G : T \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is a *guard function*.
8. $EX : A \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is an *arc expression function*.
9. $I : A \rightarrow \mathrm{Expr}_{\Sigma(\mathcal{V})}$ is an *initialisation function*.
10. $B \subseteq P \times T$ is a set of directed *inhibitor arcs*.

A modularity mechanism : place fusion

- strong constraint : requirement that the model be compositional so as to map the inductive structure of the chronicle language.
- In [CP92], definition of Modular CPN $MCPN = (S, PF, TF)$.

S. Christensen and L. Petrucci, "Towards a modular analysis of coloured Petri nets", ATPN, 1992.

- Problematics of the definition of place fusion :
  - compositional model $\longrightarrow$ fusion of MCPN instead of CPN
  - CPN Tools functionality : change of initial markings ~~and initial types~~ after a place fusion.
- Accordingly, modification of [CP92] to define MCPN, MCPN fusion, and the resulting CPN.

Outline

1st model :

- modular model
- one token in each place containing lists of recognitions
- non-deterministic : necessity to define a transition firing strategy
- non-concurrent
- no event flow management

P. Carle, C. Choppy, and R. Kervarc, "Behaviour recognition using chronicles", TASE, 2011.

P. Carle, C. Choppy, R. Kervarc, and A. Piel, "Behavioural analysis for distributed simulations", APSEC, 2012.

Design stages

2nd model : transition to multi-token nets

- modular model
- one token for each recognition
- first control structures
- beginning of some concurrency
- no event flow management

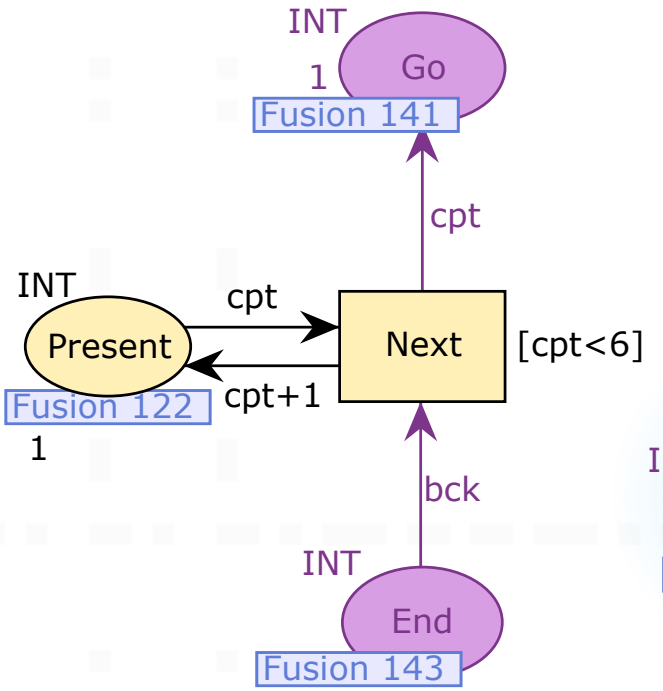
Design stages

3rd model : adding a control structure for event flow management

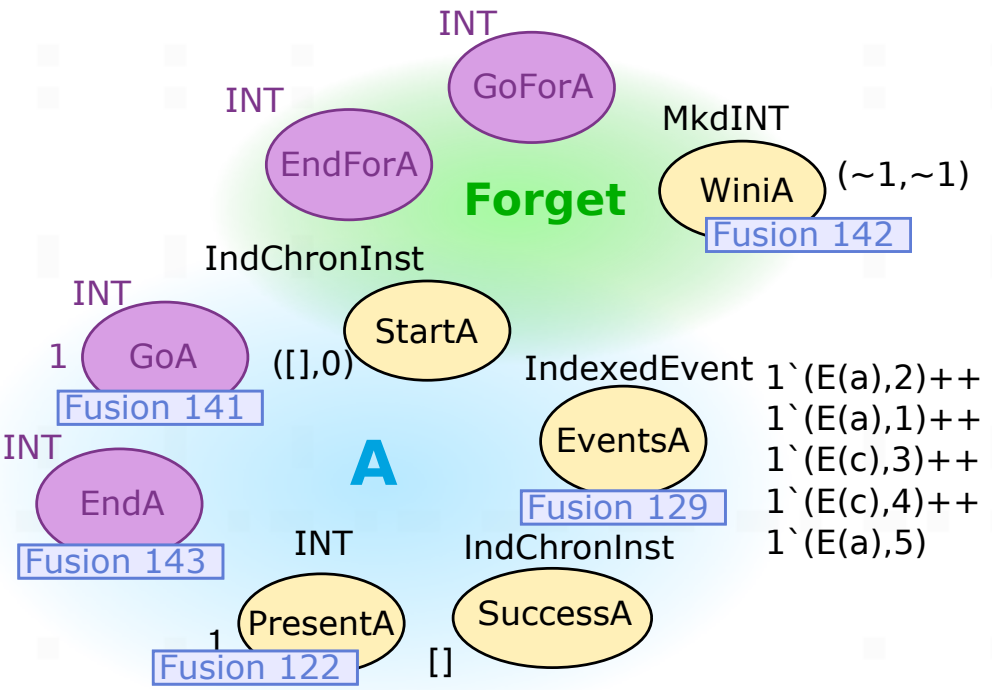
- modular model
- one token for each recognition
- deterministic model
- concurrent model
- event flow management

P. Carle, C. Choppy, R. Kervarc, and A. Piel, "A formal coloured Petri net model for hazard detection in large event flows", APSEC, 2013.

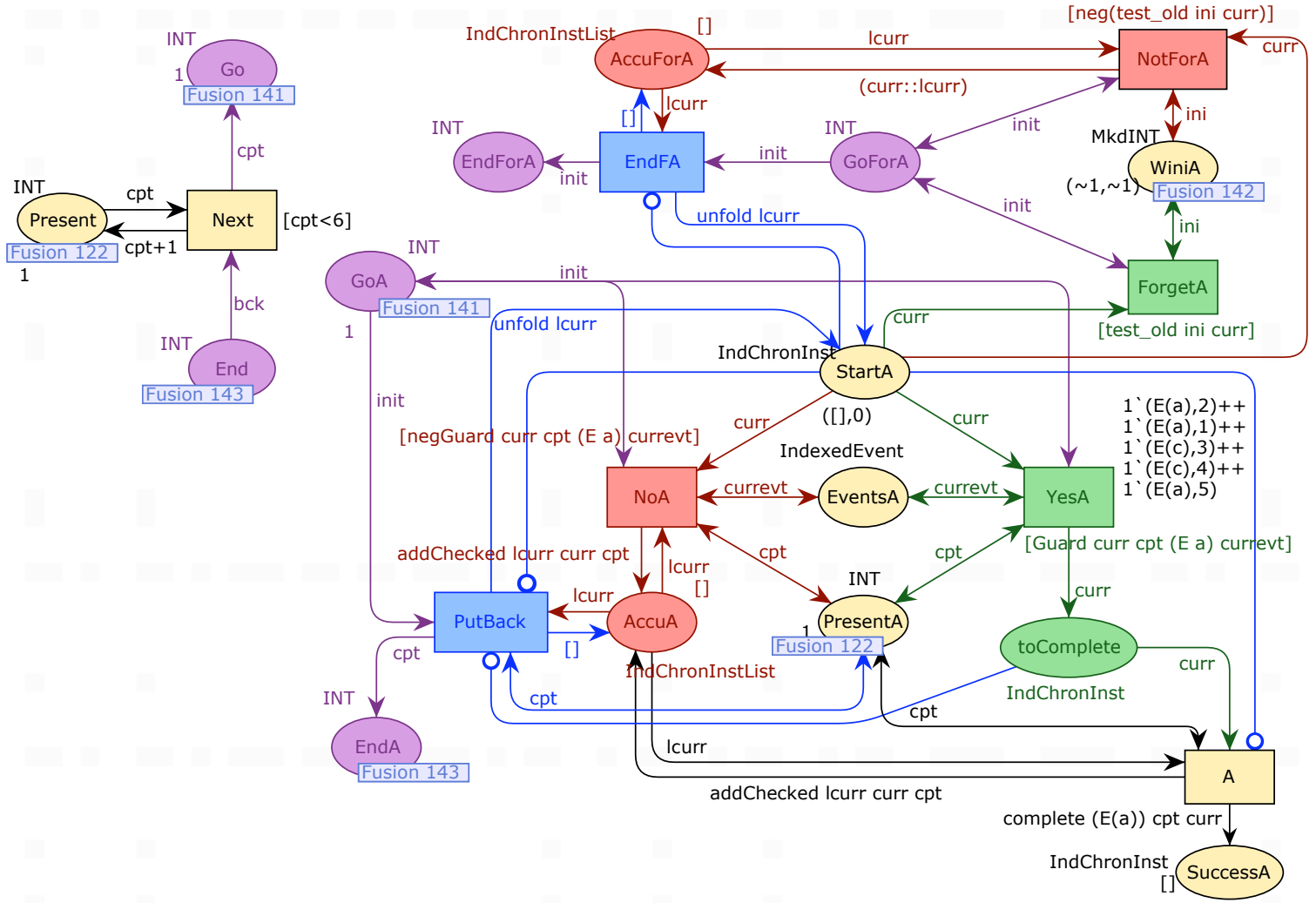
Structure of net recognising chronicle A (including the event counter)



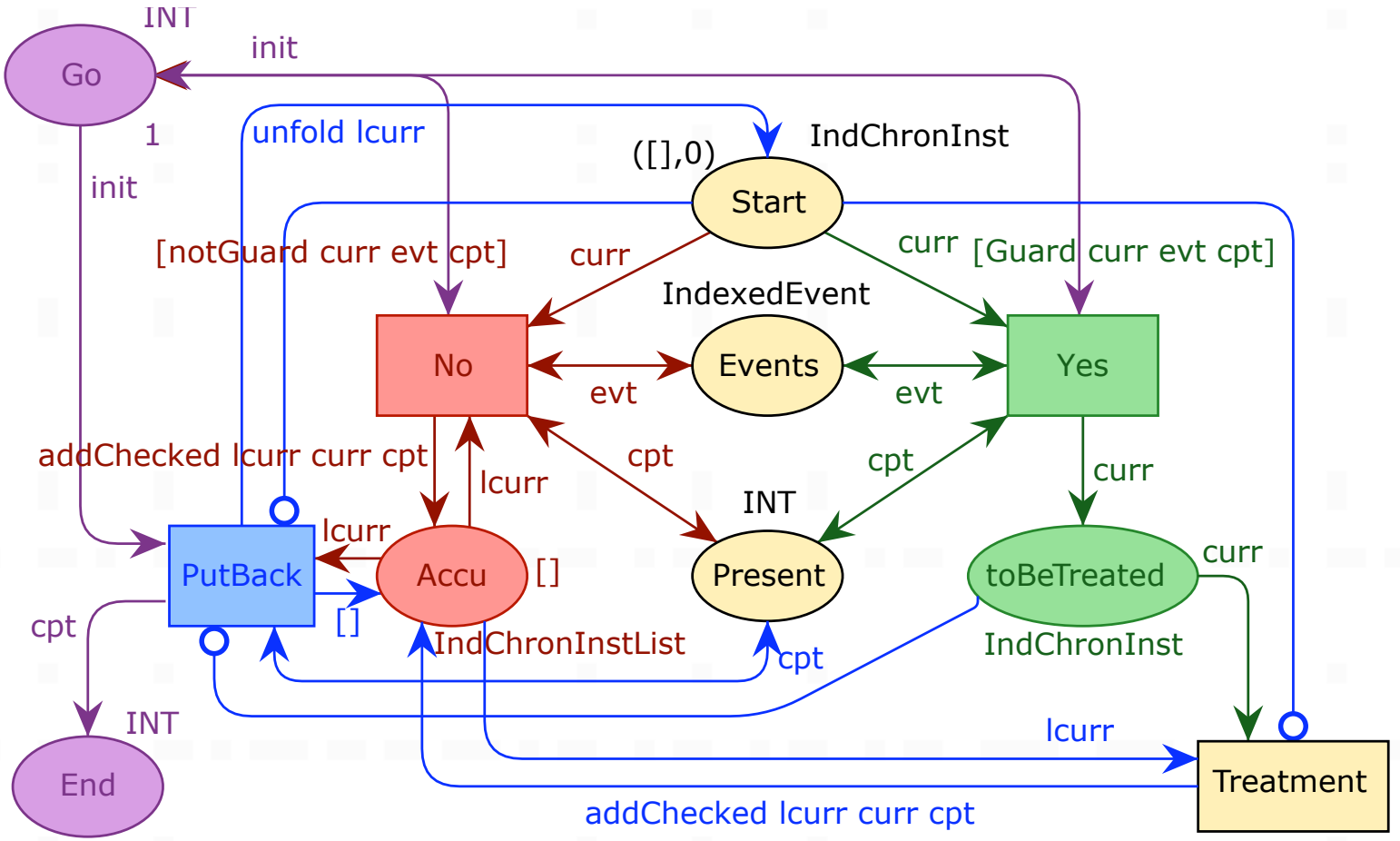
event counter (CPT)



Net recognising chronicle *A* (including the event counter)



Token splitter

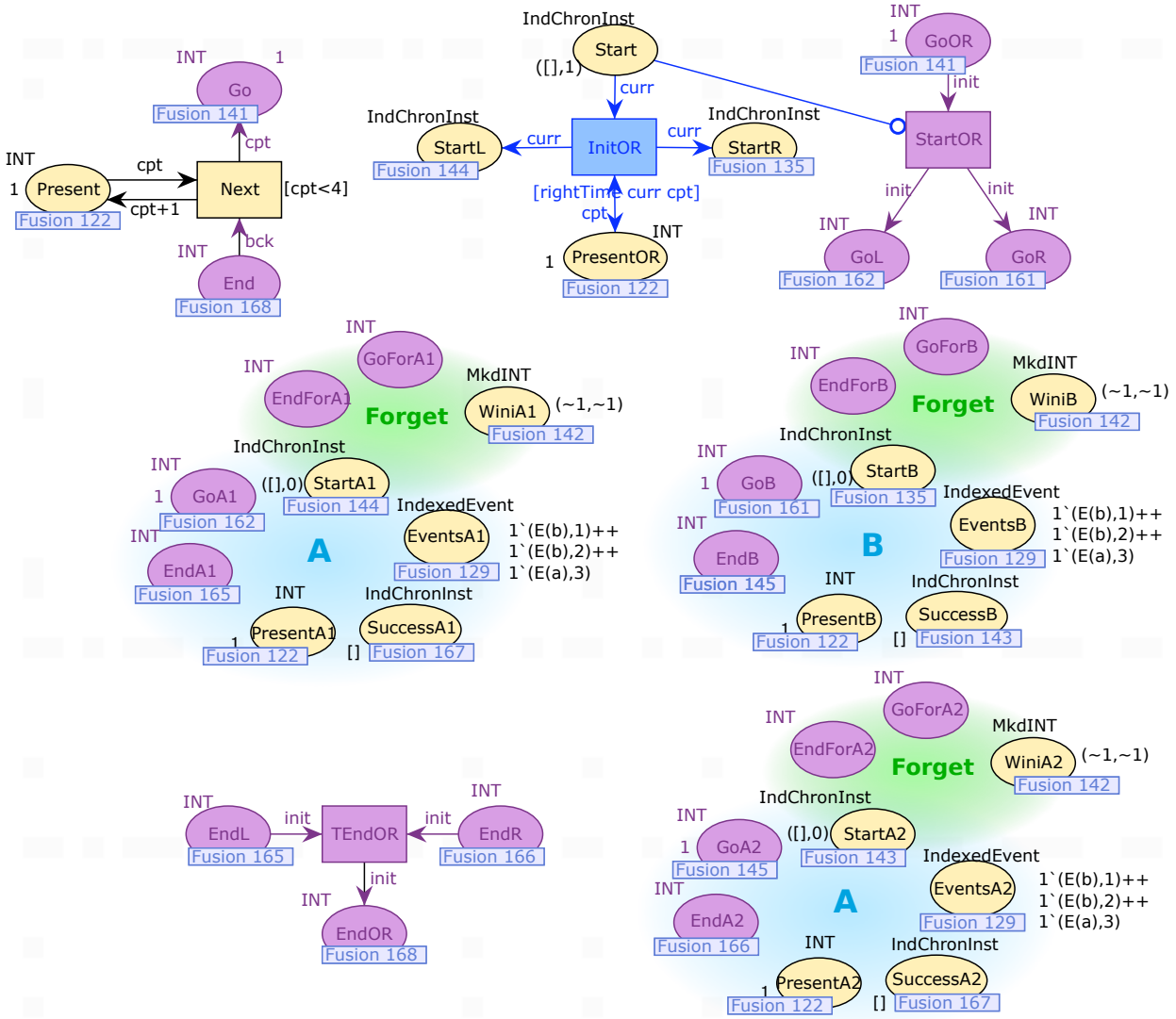


Nets will be built in 2 steps :

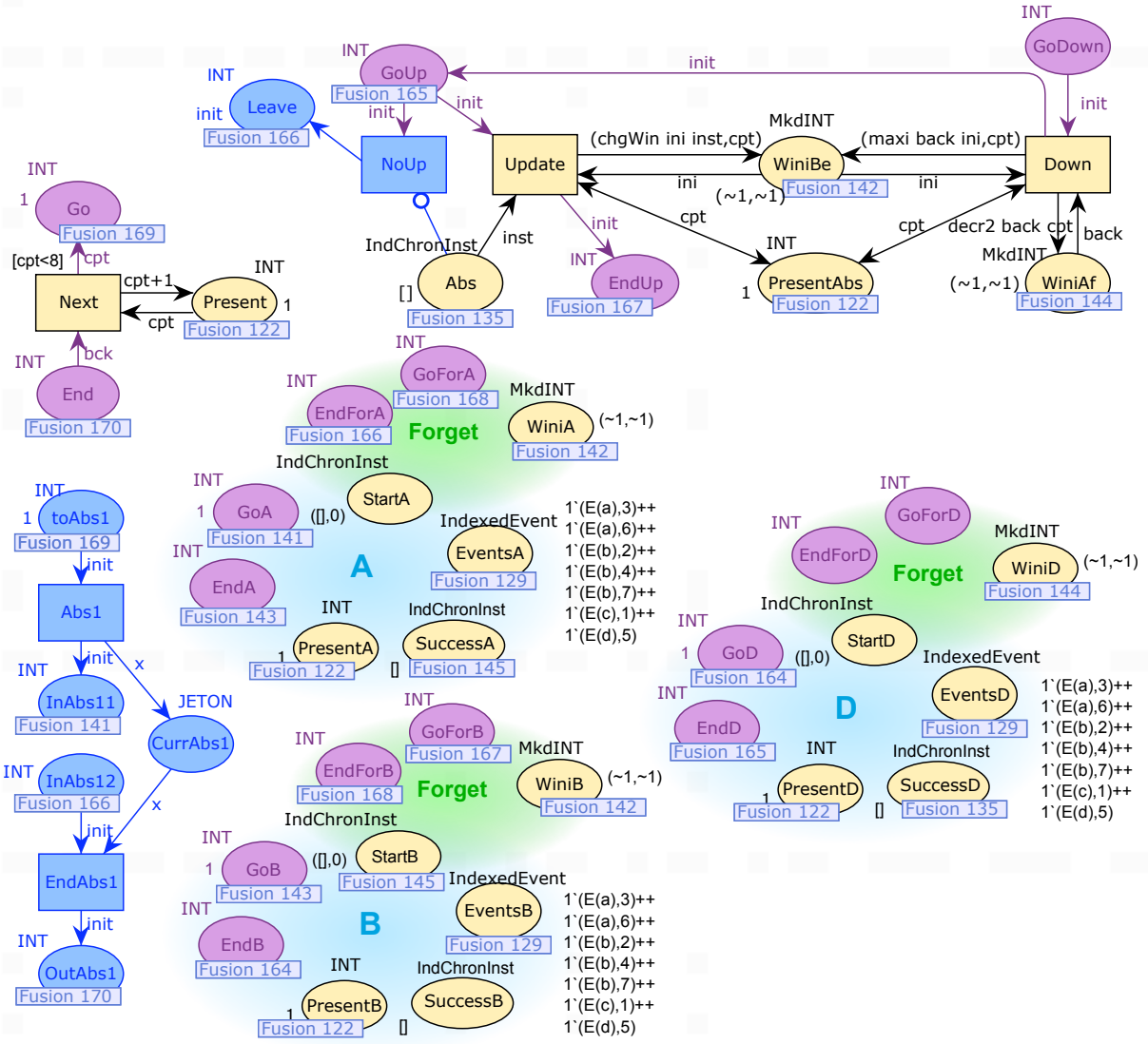
1. main mechanism of the recognition process $N'(C)$
2. fusion with the event counter CPT to build $N(C)$

$$N(C) = \mathit{Fusion}(\{N'(C), \mathtt{CPT}\}, \{(\mathtt{Go}(\mathtt{CPT}), \{\mathtt{Go}(\mathtt{CPT}), \mathtt{Go}(C)\}), (\mathtt{Present}(\mathtt{CPT}), \{\mathtt{Present}(\mathtt{CPT}), \mathtt{Present}(C)\}), (\mathtt{End}(\mathtt{CPT}), \mathtt{End}(\mathtt{CPT}), \mathtt{End}(C)\})\})$$

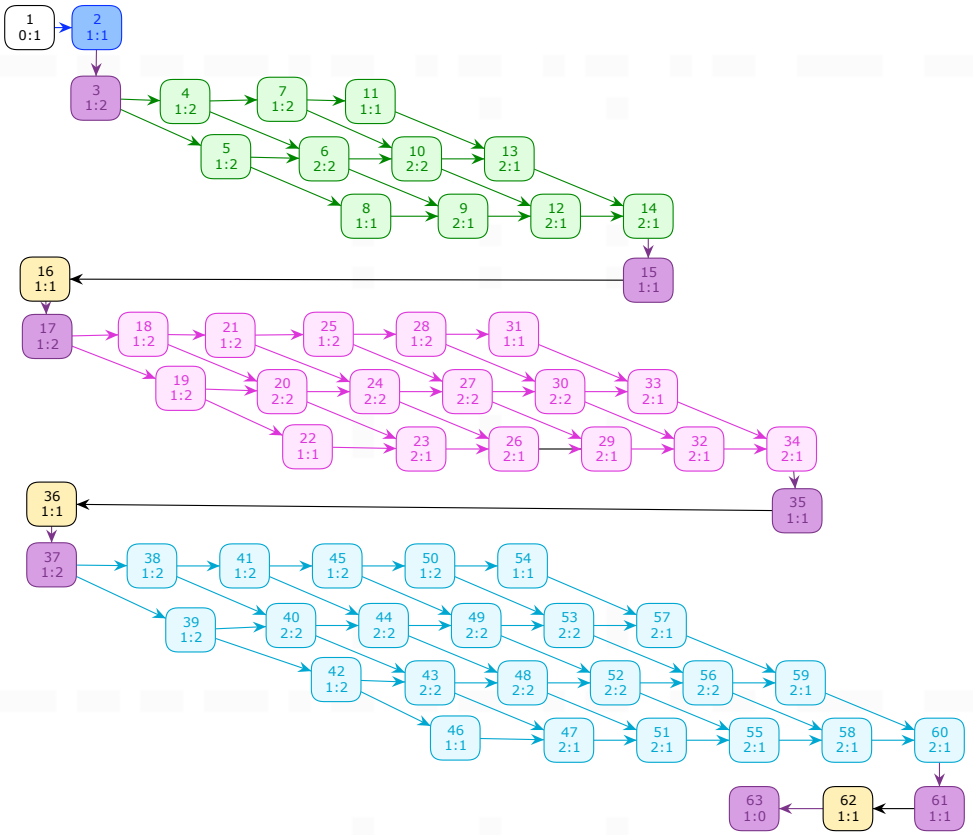
Sequence and disjunction on chronicle $A \,||\, (B\ A)$



Absence on chronicle $(A\ B) - [D]$



State Space of $N(A \,||\, (B\ A))$ on event flow $\varphi = (b, a, a)$



- regardless of the order in which the enabled transitions are fired, always the same recognitions after each event
- concurrency retained

Conclusion and perspectives

- a formal framework for behaviour recognition using CPN with inhibitor arcs and place fusion
- strong constraints on the model :
  - compositionality,
  - determinism,
  - concurrency

- extend the construction of the nets to other constructs of the chronicle language (continuous time constraints, dealing with event enriched with attributes, . . .)
- prove the adequacy of the Petri net model with the set semantics of chronicles
- put to use our Petri net behaviour recognition model for applications in the aerospace industry