Conformance verification of careflow process executions: a case study on cancer screening

Evelina Lamma, Sergio Storari
ENDIF – University of Ferrara
Federico Chesani, Paola Mello, Marco Montali
DEIS – University of Bologna
Pietro De Matteis
NOEMALIFE - Bologna
Medical guidelines and protocols

- Medical guidelines and protocols are used to improve the quality of medical care

- A medical protocol is the implementation of a guideline in a specific environment
  - Modeled as a careflow: medical actors receive objects, perform activities, operate under rules, and transmit objects to other actors
  - Components of a careflow:
    - Actors: patients, physicians, instruments, software agents
    - Objects: data, documents, images, physical samples
    - Activities: processes, actions, computations
    - Rules: constraints, conditions, limits, boundaries
Careflow conformance verification

- Careflow conformance verification to identify:
  - Wrong participant behaviors
  - Parts of the protocol not well defined
Graphical guideline editor: GOSpeL

- Simple graphical language for specifying the careflow process
- The GOSpeL representation of a careflow consists of:
  - a flow chart, which models the careflow evolution
  - a domain ontology for specifying actors, activities, and objects of the careflow
- Ontology management by using the PROTÉGÉ-2000 API
Careflow conformance verification

SCIFF language based on computational logic and abductive proof procedure (SOCS European project)
A framework for defining and verifying clinical guidelines: a case study on cancer screening

**SCIFF framework**

**Social Infrastructure**

**Protocols**

**E(evB,T)**

**Expectations**

**Reasoning**

**Verify Compliance**

**Integrity Constraints (IC): body → head**

\[ H(\text{ask}(A,B,\text{Something}), T1)) \rightarrow \]

\[ E(\text{ansyes}( B, A, \text{Something}), T2) \land T2 \geq T1 \]

\[ E(\text{ansno}( B, A, \text{Something}), T2) \land T2 \geq T1 \]
Verification module

- The SCIFF Proof Procedure:
  - processes the events: for each event it looks for a possible “unification” with the body of one (or more) SIC
  - for each IC whose “body” is verified by the events, the expectations defined in the head are generated.
- detects two types of violations:
  - \textbf{H with EN}: an actor performs activities explicitly not expected by the careflow
  - \textbf{E without H}: an actor does not act as expected by the careflow
  - \textbf{H without E}: an actor performs activities not expected by the careflow
A framework for defining and verifying clinical guidelines: a case study on cancer screening

Careflow conformance verification
SPRING project

- Joint project of the Emilia Romagna region of Italy: ENDIF – Univ. Ferrara; DEIS – Univ. Bologna; NOEMALIFE Bologna; Screening Center Bologna
- Project GOAL: to support definition and verification of cancer screening protocols
- Cancer screening to early detect and treat cancer (cervical, breast and colorectal cancers)
- Case study on cervical cancer
Careflow example

- The Lab (actor) analyzes a pap-test IDsample (object) executed on a patient Pat (actor) and sends the results PTres (object) to a physician Phy (actor). Phy evaluates IDsample as positive or negative. If positive, Phy invites (within 6 days) Pat for a treatment and a psychologist Psy (actor) invites Pat for a consultation. If negative, Phy sends a negative pap-test letter to Pat. Finally, Phy schedules the next pap-test for Pat.
Translation of W1

\[ H(\text{analysePapTest}(\text{Lab}, \text{Pat}, \text{IDSample}, \text{Phy}, \text{PTres}), \text{Tana}) \]
\[ \rightarrow \]
\[ \text{positive}(\text{PTres}) \]
\[ \land \]
\[ E(\text{treatmentInvitation}(\text{Phy}, \text{Pat}, \text{IDSample}), \text{Ttre}) \]
\[ \land \]
\[ E(\text{psyInvitation}(\text{Psy}, \text{Pat}), \text{Tpsy}) \land \text{Ttre} > \text{Tana} \land \text{Ttre} < \text{Tana} + 6 \land \text{Tpsy} > \text{Tana} \]
\[ \lor \]
\[ \text{not}(\text{positive}(\text{PTres})) \land E(\text{sendNegLetter}(\text{Phy}, \text{Pat}, \text{IDSample}, \text{PTres}), \text{Tsen}) \land \text{Tsen} > \text{Tana} \]
Example of conformant history

Happened Events

\[
\begin{align*}
H(\text{analysePapTest}(\text{lab1, pat1, 123, phy1, [results]}), 5) \\
H(\text{psyInvitation}(\text{psy1, pat1}), 7) \\
H(\text{treatmentInvitation}(\text{phy1, pat1, 123}), 10) \\
H(\text{screeningSchedule}(\text{phy1, pat1, 15apr2007}), 30)
\end{align*}
\]

Expectations

\[
\begin{align*}
E(\text{analysePapTest}(\text{Lab, Pat, IDSample, Phy, PTrRes}), \text{Ta}) \\
E(\text{psyInvitation}(\text{psy1, pat1}), \text{Tpsy}) \quad \text{Tpsy} > 5 \\
E(\text{treatmentInvitation}(\text{phy1, pat1, 123}), \text{Ttre}) \quad 5 < \text{Ttre} < 11 \\
E(\text{screeningSchedule}(\text{phy1, pat1, Date}), \text{Tsch}) \quad \text{Tsch} > 10
\end{align*}
\]
Example of violation (1/2)

Happened Events

\[
\begin{align*}
H(\text{analysePapTest}(\text{lab1}, \text{pat1}, 123, \text{phy1}, \text{[results]}), 5) \\
H(\text{psyInvitation}(\text{psy1}, \text{pat1}), 7) \\
H(\text{treatmentInvitation}(\text{phy1}, \text{pat1}, 123), 15)
\end{align*}
\]

Expectations

\[
\begin{align*}
\text{E(\text{analysePapTest}(\text{Lab}, \text{Pat}, \text{IDSample}, \text{Phy}, \text{PTRes}), \text{Ta})} \\
\text{E(\text{psyInvitation}(\text{psy1}, \text{pat1}), \text{Tpsy})} \quad \text{Tpsy} > 5 \\
\text{E(\text{treatmentInvitation}(\text{phy1}, \text{pat1}, 123), \text{Ttre})} \quad 5 < \text{Ttre} < 11
\end{align*}
\]

Violation of the time constraint
Example of violation (2/2)

Happened Events

\[ H(\text{analysePapTest}(\text{lab1}, \text{pat1}, 123, \text{phy1}, [\text{results}]), 5) \]

\[ H(\text{sendNegLetter}(\text{Phy1}, \text{Pat1}, 123, [\text{res1}, \ldots, \text{resn}]), 10) \]

The protocol evaluates the pap-test as positive but the physician as negative and behaves as negative.

The physician performs an activity not expected by the careflow.

These Expectations are not fullfilled.

Expectations

\[ E(\text{analysePapTest}(\text{Lab}, \text{Pat}, \text{IDSample}, \text{Phy}, \text{PTRes}), \text{Ta}) \]

\[ E(\text{psyInvitation}(\text{psy1}, \text{pat1}), \text{Tpsy}) \quad \text{Tpsy} > 5 \]

\[ E(\text{treatmentInvitation}(\text{phy1}, \text{pat1}, 123), \text{Ttre}) \quad 5 < \text{Ttre} < 11 \]

Start

Generated by IC1 supposing positive([results]) = true
Screening careflow model in SPRING
Traslation of the careflow model

- The careflow model is translated in 14 ICs
- executeExam translation:
  - H(eseguiEsame(TipoEsame,IdEsame),Tesa) ∧ analisi_esterna(TipoEsame)
    \(\rightarrow\) E(invioCampione(TipoEsame,IdEsame),Tinv) ∧ Tinv > Tesa.
  - H(eseguiEsame(TipoEsame,IdEsame),Tesa) ∧ analisi_interna(TipoEsame)
    \(\rightarrow\) E(invioRisultato(TipoEsame,IdReferto,Esito), Tris) ∧ Tris > Tesa.
  - H(invioCampione(TipoEsame,IdEsame),Tinv)
    \(\rightarrow\) E(invioRisultato(TipoEsame,IdReferto,Esito),Tris) ∧ Tris > Tinv.
Screening event log

- Database of the screening center translated in event log
- Some incorrect behaviours have been randomly introduced in the event log
- The resulting event log consists of 1950 careflow process executions:
  - Shortest careflow process execution consists of one event (the invitation to take part to the screening followed by no response)
  - Longest careflow process execution consists of 18 events (representing the whole careflow plus the repetition of some laboratory exams due to an undecidable analysis result).
  - The average number of events is 4
Conformance verification results

- Conformance verification execution time:
  - 30 min Total and 1 sec Average

- Conformance result:
  - 877 Conformant executions over 1950

- Analysis of non conformant careflow process executions:
  - Executions classified as conformant were confirmed
  - Some particular executions were erroneously classified as non conformant:
    - We introduced some special abducibles in the ICs:
      - To classify these executions as conformant
      - To warn about special executions

- Second verification round: 64 executions are still not conformant (“wrong behaviour” introduced in the database and some insights)
Conclusions

- Use computational logic to verify conformance of participant behaviors within a careflow

- Our approach proposes:
  - Formal language to model the careflow
  - Abductive proof procedure to verify the conformance

- Case study on cervical cancer screening
Future works

- Change GOSpeL with another graphical guideline modeling notation:
  - GLARE: joint work with Terenziani/Bottrighi
  - ASBRU
  - Etc..

- Use gSCIFF:
  - Properties verification
Thank you!
A framework for defining and verifying clinical guidelines: a case study on cancer screening

References

Publications


Web references:

- The SCIIF Abductive Proof Procedure: [http://www-lia.deis.unibo.it/research/sciff/](http://www-lia.deis.unibo.it/research/sciff/)
A framework for defining and verifying clinical guidelines: a case study on cancer screening
Translation of W2

\[ H(\text{sendNegLetter}(\text{Phy}, \text{Pat}, \text{IDsample}, \text{PTres}), \text{Tsen}) \]

\[ \rightarrow \]

\[ E(\text{screeningSchedule}(\text{Phy}, \text{Pat}, \text{InvDate}), \text{Tscr}) \land \text{Tscr} > \text{Tsen} \]
A framework for defining and verifying clinical guidelines: a case study on cancer screening

Translation of W3

\[ H(treatmentInvitation(Phy,Pat,IDsample),Ttre) \land H(psyInvitation(Psy,Pat),Tpsy) \rightarrow E(screeningSchedule(Phy,Pat,InvDate),Tscr) \land Tscr > Ttre \land Tscr > Tpsy \]
GOSpeL graphical elements

<table>
<thead>
<tr>
<th>Activities</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>atomic activity</td>
<td>complex activity</td>
<td>iteration</td>
<td>while</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gateways</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>exclusive choice</td>
<td>deferred choice</td>
<td>parallel fork</td>
<td>parallel join</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Start Blocks</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>cyclic start</td>
<td>return</td>
<td>end</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PRIN meeting - Bologna 30/01/2007
A framework for defining and verifying clinical guidelines: a case study on cancer screening
Some GOSpeL blocks can be mapped into events (event-blocks):

- Activity blocks represent events specified by:
  - name of the associated ontological activity
  - variables representing formal participants
  - Example: `hold(Phy,Pat)`

- START / END blocks represent special events

Each translation starts from an event-block A:

- This event A is supposed to happen
- The relations after A in the model describe what it is expected to happen after A
Idea behind the translation (2/2)

- The meaning of a translation in a SIC:
  - The body represents that an event is happened
  - The head represents what the model prescribe to happen after this event
- The model is translated in a set of SIC

- Two issues:
  - Isolate a part of the model that can be mapped in a SIC
  - Build a recursive translation algorithm
Definition of Minimal windows

- A group of contiguous blocks
- Properties:
  - Window source and fringe must contain only event-blocks
  - Inside the window there must be only split and/or merge blocks (minimal)
  - All the outgoing (ingoing) relations exiting from (going to) a split block (merge block) must be considered
- Each minimal window is translated into a SIC
Translation of GOSpeL
Translation of a minimal window

- Events in the window source became $H$
- Events in the window fringe became $E$
- Gateways inside the window contribute to the SIC structure:
  - Deferred choice: a disjunction of events is inserted in the head
  - Exclusive choice: a disjunction of events is inserted in the head and a logic condition is associated to each alternative flow
  - Parallel split: a conjunction of events is inserted in the head
  - Parallel join: a conjunction of events is inserted in the body
GPROVE framework architecture