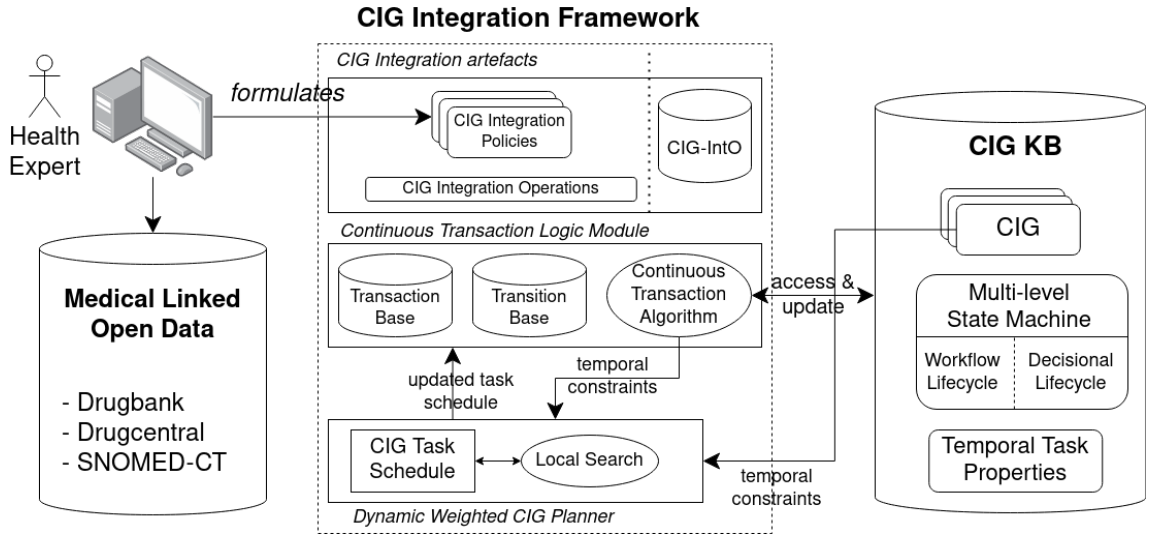
Solution Report

Team: William, Raza, Samina

Part 1: Architecture and use

Architecture

Please provide a diagram illustrating the system architecture and briefly explain its components. (**source**: [1])



Health experts identify comorbid considerations (e.g., drug-drug, drug-condition interactions) aided by the *Medical Linked Open Data* tool (MLOD), which includes semantic sources such as Drugbank and Drugcentral. Subsequently, they formulate suitable *CIG integration policies*, in terms of the *CIG Integration Ontology* (CIG-IntO). These policies apply *CIG integration operations* on CIG tasks in order to resolve the comorbid consideration.

The *Continuous Transaction Logic Module* is a logics-based module that represents *execution-time* CIG integration semantics with a *transactional flavour*. At execution-time, the patient's health profile will continuously evolve, often in unforeseeable ways: by analogy, our method continuously refines comorbid CIG integration, applying new operations when all conditions hold, and, inversely, rolling back prior operations when a condition no longer holds. The module keeps a set of transactions, *Transaction Base*, and a set of elementary transitions, *Transition Base*, which collectively implement *CIG integration policies* and *CIG integration operations*. The *Continuous Transaction Algorithm* [2], based on a RETE network, implements a continuous version of Transaction Logic [3].

CIG integration policies are informed by the *Workflow Lifecycle* and *Decisional Lifecycle* of CIG tasks, part of the *Multi-level State Machine* (CIG KB). The *Workflow Lifecycle* keeps the current workflow state of a CIG task (e.g., active, started completed). The *Decisional Lifecycle* (CIG KB) captures the uncertainty that will exist at execution-time, i.e., whether the particular task will be executed based on prior decisions in the workflow. Indeed, some integration operations are needed even when it is still uncertain whether a task will be executed: when the task ends up being part of a non-chosen decisional branch, these integration operations may have to be rolled back.

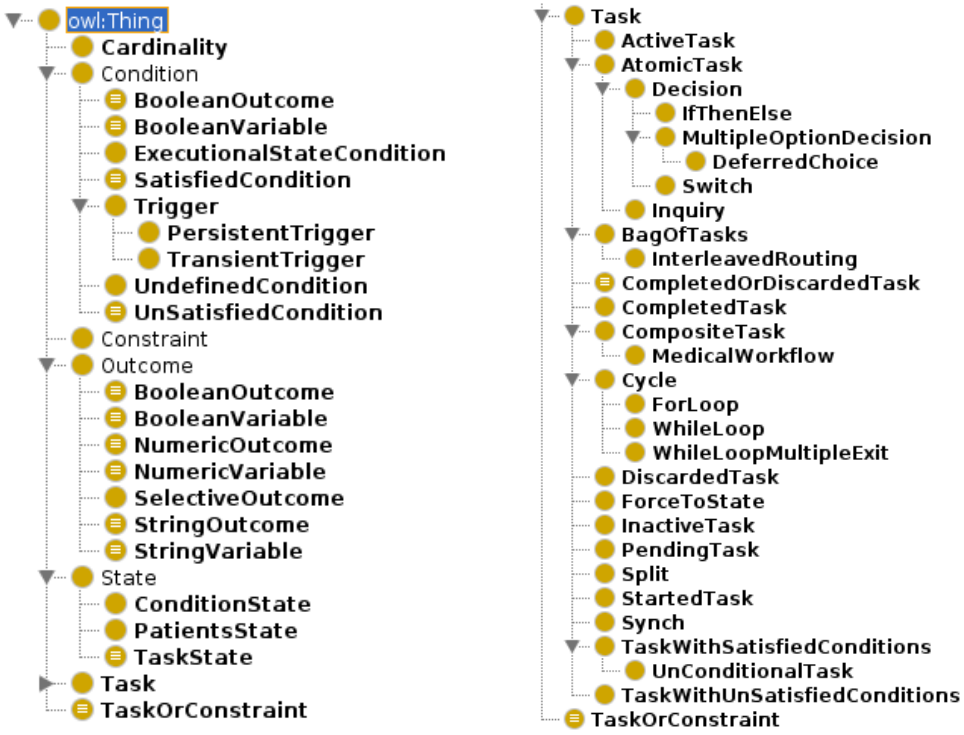
State operations will be applied on the *Multi-level State Machine* (*Workflow Lifecycle*) in the CIG KB – a conflict detection scheme [4] flags potential conflicts between state operations. Temporal constraints

can be issued by CIG integration policies, and other temporal constraints will be inherent in the CIG (e.g., sequential tasks); these are passed to the *Dynamic Weighted CIG Planner*. At execution-time, this component utilizes the current set of (weighted) temporal constraints to find a globally optimal CIG task schedule. This task is represented as an optimization problem and solved using a Hill Climbing algorithm with an objective function. Based on the optimal CIG task schedule, (fuzzy) truth values are associated with temporal constraints. When these truth values lie beneath a threshold, the temporal constraint will be considered unsuccessful, and the transaction will be rolled back.

CIG representation

Please explain the formalism used to represent CPGs.
(**source**: [5], Chapter 4)

Based on the seminal works by Mulyar et al. [6] and Peleg et al. [7], we developed the CPG Domain Knowledge Ontology, or CIG ontology, to represent the identified workflow and decisional patterns as OWL classes, properties and instances. We show the full class hierarchy below:



Below, we summarize some of the workflow and decisional patterns:

Control Flow

- *Sequence*: tasks can be executed in simple sequential order.
- *Preconditions*: if a task's pre-condition is not satisfied, then the task will not be executed.
- *Nesting*: a composite task can be defined, which is composed of several sub-tasks. Once all the sub-tasks are completed, the composite task is completed as well.
- *Parallel split*: several paths can be executed in parallel.
- *Exclusive choice*: similar to a parallel split, but only one of the paths will be selected for execution based on a selection mechanism.

- *Multiple choice*: similar to a parallel split, but only some of the paths will be selected for execution based on a selection mechanism.
- *Synchronization point*: parallel paths converge at this point to a single path of execution; when *all* incoming paths are completed, the next task after this point will be activated.
- *Simple merge*: similar to a synchronization point, but completion of *any* incoming path will cause the next task after the merge to be activated. Execution of the other incoming paths continues.
- *Structured N-out-of-M join*: similar to a synchronization point, but completion of *n out of m* parallel paths will cause the next task after the merge to be activated. Execution of the other paths continues.
- *Cancelling discriminator*: similar to a synchronization point, but when *any* path reaches the cancelling discriminator, execution of the rest of the paths stops.
- *Cancelling N-out-of-M join*: similar to a structured N-out-of-M join, but when *n out of m* branches reach the join, execution of the other paths stops.
- *Arbitrary cycle*: repetitive tasks that are iterated for a predefined number of times, or until a specific condition is satisfied. A cycle may have several entry and exit points.
- *Milestone*: a task that is activated only after a first task has been executed, and before a third task is executed. (Can be considered a special type of precondition, using executional states of tasks.)
- *Triggers*: conditions that may trigger tasks. E.g., having a BP value over a threshold can trigger task “prescribe hypertension medication” (which may have pre-conditions). *Transient* triggers are lost if they are not acted upon instantly (e.g., because a pre-condition is currently not met); *persistent* triggers will last until they are acted upon by a task.
- *Critical section*: a critical section includes multiple composite tasks; while a sub-task of a composite task is being executed, none of the sub-tasks of the other composite tasks can be activated.
- *Interleaved parallel routing*: a composite task consisting of a set of partially ordered tasks.
- *Interleaved routing*: a composite task including a set of tasks to be executed only once, in any order. No two sub-tasks can be executed at the same time.

Decisional Patterns

- *If-then-else*: if a specific condition is satisfied, the "then" part of the if-then-else is executed; otherwise, the "else" part will be executed.
- *Switch*: this construct has an arbitrary number of conditions, each with an associated task: if a condition is satisfied, the associated task will be executed; the remaining tasks are discarded. (If two or more conditions are satisfied concurrently, the higher-priority task will be chosen.)
- *Argumentation rules*: these represent the "pros" and "cons" of the different decisional options. An argumentation rule will increment or decrement an option's score; either a single option with the highest score will be selected, or all options with scores above a threshold will be selected (this can be an overall threshold, or specific to the particular option).
- *Preference for options*: users can express different preferences for decision options; these can be shown to the users, to leave the decisions to them, or they can be acted on by an execution engine.

Domain knowledge representation

If additional domain knowledge is required, please explain how it is represented. Indicate whether standards (e.g., SNOMED-CT, FHIR, standard domain ontologies) are being utilized.
(**source**: [6])

We provide a Medical Linked Open Data (MLOD) tool [7], which integrates multiple semantic knowledge sources, to identify comorbid considerations (e.g., drug-drug, drug-condition interactions) within multiple comorbid CIG. While relevant sources are freely available, such as DrugBank[1], DrugCentral[2] and SNOMED-CT [8], these exist in custom formats (XML, RDB, CSV) and are not interlinked. This limits the development of real-world solutions for detecting and resolving adverse interactions. To fill this gap, we provide a semantic integrated MLOD together with means for querying and updating this MLOD. The MLOD currently includes DrugBank and DrugCentral (using Bio2RDF[3] as common drug vocabulary), and SNOMED-CT.

E.g., MLOD encodes the drug-drug-interaction between Erythromycin and Warfarin as follows:

```
dv:drug:6964 dc:title "Erythromycin"@en ; dv:drug-interactor b2r:drug_interaction:868 .
dv:drug:5376 dc:title "Warfarin"@en ; dv:drug-interactor b2r:drug_interaction:868 .
b2r:drug_interaction:868 a dv:Drug-Drug-Interaction ;
      dc:description "ERYTHROMYCIN/WARFARIN [VA Drug Interaction]" ; dv:interaction-risk "Critical" .
```

E.g., MLOD captures a DCI between Levetiracetam and condition “Kidney disease” as follows:

```
b2r:drug:4194 dc:title "Levetiracetam"@en ; dv:drug-interactor b2r:drug_interaction:151538 .
b2r:drug_condition:21002292 dc:title "Kidney disease"@en ;
      dv:condition-interactor b2r:drug_interaction:151538 .
b2r:drug_interaction:151538 a dv:Drug-Condition-Interaction .
```

We attach CIG tasks to their prescribed medication (if any) using the SNOMED-CT OWL ontology. For instance (“sct” represents the SNOMED namespace):

```
:prescribe_Epilepsy_LEV a :CIGTask ; involves :presc_proc_Epilepsy_LEV .
:presc_proc_Epilepsy_LEV a sct:Prescription; sct:drugUsed sno:LEV .
```

These annotations enable automated identification of equivalent drug tasks across comorbid CIG, as well as drug-drug and drug-condition interactions.

Mode of use

Please explain the intended mode of use of the system: who are the intended end-users, when is the system to be used: during patient encounter, real-time vs. simulation, etc.
(**main source**: [1])

The system is intended to be utilized in real-time, as new health-related data becomes available – test results from laboratory information systems, specialist data from electronic medical records, or patient-reported outcomes during an encounter. In response to this new health data, the decisional state of comorbid CIG workflows may change, i.e., certain options will be followed at decision nodes. In this new decisional state, certain CIG integration operations will become relevant and need to be applied; other, prior operations may have to be rolled back, as they are no longer safe nor efficient. At execution-time, the clinician will be notified of any applied or rolled-back CIG integration operations, and may choose to over-rule them.

[1] http://www.drugbank.com/
[2] http://drugcentral.org/
[3] http://bio2rdf.org/

With regards to individual CIG execution, the system will execute CIG based on the new health data; this will result in some tasks being activated, discarded, or completed. The activation of a task means it is next in line for execution: at this point, the clinician will similarly be notified, and may choose to start the task.

Strengths of the approach

Does the approach have very good support for particular features? Which? Please justify. What is the singular point of strength of your approach?

(**main source**: [1])

- As CIG workflows progress at execution-time, more data will become available (e.g., test outcomes) and real-time events will take place (e.g., exacerbations): i.e., the patient's health profile will continuously evolve. By analogy, comorbid CIG integration should be continuously refined as well. This necessitates *dynamic, execution-time decision-making*, which is formalized using Transaction Logic ensuring an "all-or-nothing" flavour: (re-)applying an integration decision when all health-related conditions hold, and rolling back integration decisions when a condition no longer holds.

- In clinical practice, comorbid conditions often need to be dealt with in complex and nuanced ways, depending on dynamic health parameters. CIG integration policies can be outfitted with elements that are conditional on the patient's evolving health profile, including entry- and exit-conditions, and relevant to specific point in time (e.g., occurrence of another CIG task).

- A range of temporal constraints will restrict CIG workflows, such as sequential relations and clinically safe delays; when performing CIG integration, temporal constraints are often issued to guarantee clinical safety or efficiency (e.g., by avoiding temporal overlaps). A *dynamic and weighted planning system* reconciles all temporal constraints at execution-time, and continuously updates an optimal plan schedule. We provide a set of temporal constraints for a variety of use cases, which can be weighted (reflecting their importance) and have fuzzy truth values (reflecting degree of adherence).

- During our work, we observed a lack of ready-to-use, up-to-date, and integrated semantic knowledge for the detection of drug-drug and drug-illness interactions. We provide a set of integrated semantic knowledge sources, called *Medical Linked Open Data (MLOD)* [7], which can be leveraged to (semi-)automatically identify adverse interactions within comorbid CIG. MLOD includes state-of-the-art drug interaction knowledge sources, including DrugBank[4] and DrugCentral[5].

Part 2: Features

Section A outlines a set of features that relate to possible interactions among advice offered by CPGs.
Section B lists a set of features that relate to possible mitigation strategies for these interactions.
Section C lists other possible features. We include a brief example to illustrate each feature.
For each of the features, please indicate whether it is supported, and, if so, briefly explain how.

Section A. Interactions among CPGs' advice

A1: Drug from a CPG has an effect on a comorbid condition

[4] http://www.drugbank.com/
[5] http://drugcentral.org/

For example, low-dose Aspirin (Cardiovascular Disease CPG) affects Duodenal Ulcer (comorbid condition).

Implemented (Y/N): Y
Brief description: The MLOD tool will detect the particular drug-condition interaction, if found in Drugbank or Drugcentral. Subsequently, the clinician formulates a CIG integration policy that will cope with the interaction – e.g., a *ReplaceTasks Policy* [1]. At execution-time, the integration policy will coordinate comorbid CIG integration to avoid the adverse interaction – e.g., replacing the aspirin with a safer alternative.

A2: Two or more drugs from different CPGs interact
For example, antibiotics such as Trimethoprim/Sulfamethoxazole impact the anticoagulant effect of Warfarin.

Implemented (Y/N): Y
Brief description: The MLOD tool will detect the particular drug-drug interaction, if found in Drugbank or Drugcentral. Subsequently, the clinician formulates a CIG integration policy that will cope with the interaction at execution-time.

A3: Clinical goals from different CPGs conflict
For example, the goal of preventing thrombosis conflicts with the goal of preventing bleeding during surgery.

Implemented (Y/N): Partially
Brief description: A CIG integration policy can resolve this type of comorbid consideration – e.g., an *Event Conditional Replace Policy* that will replace a drug treatment with replacement tasks *before* and *after* the surgery. However, there does not exist a concept of goals in our approach, and the clinician would have to manually target the related tasks. It would also not be possible to automatically detect this comorbid consideration using the MLOD tool.

A4: Conflicting actions (e.g., drugs, procedures) from different CPGs

For example, one CPG recommends administration of Clopidogrel (Transient Ischemic Attack CPG) while another recommends suspending Clopidogrel (Coronary Artery Bypass Grafting CPG).

Implemented (Y/N): Y
Brief description: By annotating CIG tasks with SNOMED-CT terms, our MLOD tool can detect these types of tasks (i.e., subscribing/suspending the same drug)[6]. Subsequently, the clinician may formulate a CIG integration policy to resolve the issue – e.g., a *Separate Tasks Policy* that will avoid temporal overlap between the two tasks.

A5: Duplicate or redundant advice from different CPGs
For example, Calcium Channel Blockers are recommended in Hypertension and Cardiovascular Disease CPGs.

[6]Note that this detection would not consider the conflicting nature of the actions, but merely that the tasks involve the same drug.

Implemented (Y/N): Y
Brief description: By annotating CIG tasks with SNOMED-CT terms, our MLOD tool can detect these types of tasks (i.e., prescribing the same drug). Subsequently, the clinician may formulate a CIG integration policy to resolve the issue – e.g., a *Redundant Tasks Policy* that will discard a redundant task if deemed clinically safe.

A6: Temporal relationship between different CPGs
For example, take Cefpodoxime (Acute Otitis Media CPG) two hours after taking antacids (Gastroesophageal Reflux Disease CPG).

Implemented (Y/N): Y
Brief description: If applicable, the MLOD tool can detect the particular drug-drug interaction. Subsequently, the clinician can formulates a CIG integration policy that will cope with the interaction at execution-time – e.g., a *Separate Tasks Policy* [1] to avoid a temporal overlap.

A7: Multiple interactions from different CPGs interacting at the same time
For example, replacing low-dose Aspirin (Transient Ischemic Attack CPG) with Proton Pump Inhibitor to mitigate Duodenal Ulcer (Duodenal Ulcer CPG) impacts new comorbid condition of Osteoporosis (Osteoporosis CPG).

Implemented (Y/N): Y
Brief description: Coping with multiple interactions is not a problem – see Case 1 for an example.

Section B. Mitigation strategies when CPGs offer interacting advice

A mitigation strategy is an action taken to address one or many of the interactions that were identified above.

B1: Adding a drug to mitigate an adverse effect
For example, add a PPI to mitigate the Duodenal Ulcer due-to Aspirin.

Implemented (Y/N): Y
Brief description: The clinician can formulate CIG integration policies that may replace or discard existing tasks, and insert new tasks. We note that the choice of PPI would be made by the clinician themselves, e.g., based on recommendations from the clinical guideline; this is not automated.

B2: Adjust drug dosage
For example, a reduction of 10% of warfarin dosage.

Implemented (Y/N): Y
Brief description: The clinician can formulate a *Replace Tasks Policy* that replaces an existing task with the essentially same task (e.g., warfarin prescription) but different properties (e.g., dosage). See Case 3 for an example.

B3: Monitor the effect of a drug
For example, monitor progression of the Duodenal Ulcer during overlapping treatment with Aspirin.

Implemented (Y/N): Y
Brief description: The clinician can formulate an *Event Conditional Replace Policy*, which will be triggered when an event takes place, such as a drug prescription or comorbid condition (e.g., Duodenal Ulcer). During the event, the policy can replace a particular task (e.g., Aspirin prescription) when a certain condition holds; in this case, an exacerbation of the Duodenal Ulcer.

B4: Replacing a drug with a safer / non-interacting drug / more effective drug for comorbidity
For example, replace Aspirin with Clopidogrel for a patient with Duodenal Ulcer.

Implemented (Y/N): Y
Brief description: A *Replace Tasks Policy* can be formulated to replace a given task with another task at execution-time. As before, the choice of Clopidogrel would be made by the clinician themselves, e.g., based on recommendations from the clinical guideline; this is not automated.

B5: Discard unsafe/interacting drug
For example, suspend ACE inhibitor when eGFR value drops by over 30% over 4 months.

Implemented (Y/N): Y
Brief description: In general, a *Discard Tasks Policy* can be formulated to discard a CIG task. For this example, the clinician can formulate an *Event Conditional Replace Policy* that is triggered during (for instance) Chronic Kidney Disease. During this event, the policy will suspend or replace the ACE inhibitor when a certain condition holds, i.e., when the eGFR value drops by the specified amount.

B6: Delay a task to avoid a temporal overlap
For example, stop Dabigatran 4 days prior to surgery for a patient with high bleeding risk.

Implemented (Y/N): Y
Brief description: a *Separate Tasks Policy* can be formulated to avoid an overlap between CIG tasks. This policy can issue a temporal constraint that will delay the Dabigatran treatment until after completion of the surgery with an arbitrary separation period (e.g., 4 days).

B7: Add a task to ensure a temporal overlap
For example, for a patient with high risk of thromboembolism who is undergoing surgery with a high risk of bleeding, suspending Warfarin 5 days prior a surgery and resuming it one day after the surgery, introduces a 6-day period where the patient is at risk of bleeding; bridge with heparin starting on day 3 prior to surgery till the day of surgery to ensure overlap of the surgery context and the thromboembolism prevention context.

Implemented (Y/N): Partially (extension needed)
Brief description: This required a generic version of the *Event Conditional Replace Policy,* called the *Event Conditional Policy*, which includes (a) a more generic "relative" element (before, only "during" and "after" were available), and (b) support for arbitrary operations, aside from replacements.

A first *Event Conditional Policy* can be formulated that is triggered by the surgery CIG task (event). The "relative" element will discard the warfarin treatment, with *before* property of 5 days, and *after* property of 1 day – i.e., the discard operation will only be applied during this time span relative to the

surgery. Then, a second *Event Conditional Policy* can be formulated, with a "relative" element that adds the heparin bridge therapy, with a *before* property of 3 days – i.e., this add operation will only be applied during the 3 days before the surgery.

B8: Are there any other mitigation strategies for the multimorbidity CPG problem that you have implemented?

Implemented (Y/N): Y
Brief description: A *Simultaneous Tasks Policy* can be formulated to *ensure* a temporal overlap between CIG tasks, as opposed to *avoiding* a temporal overlap. See [1], Section 2.4.2.

The *Redundant Tasks Policy* is outfitted with temporal features: this is meant for cases where a redundant task may only be discarded when the results of another "essential" task are still valid (i.e., their "validity period" has not yet been exceeded). E.g., the CPG for Chronic Obstructive Pulmonary Disease (COPD) and Pulmonary Embolism (PE) respectively recommend X-rays/CT-scans and CT-PA scans, whereas CT-PA scans may be utilized for both COPD and PE diagnosis. Here, the CT-PA scan represents the "essential" task, and X-rays/CT-scans are the redundant tasks. CT-PA scan results have a limited validity period, depending on the evolution of the illness; in case this validity period is exceeded, the X-rays/CT-scans will still need to take place. See [1], Section 2.4.4.

Section C. Other features

C1: Patient preferences and/or patient burden
For example, choosing one drug over another due to lower price; or choosing DOACs over warfarin to avoid checking INR on regular basis.

Implemented (Y/N): N
Brief description: There does not exist a concept of "goals" in our approach, such as reducing treatment costs or adherence to patient preferences. Hence, a clinician would have to manually implement such goals as CIG integration policies on the relevant tasks (e.g., replacing a more expensive drug with a cheaper one, using a *Replace Tasks Policy*).

C2: Optimization of clinical resources
For example, grouping tests on the same day.

Implemented (Y/N): Y
Brief description: A *Simultaneous Tasks Policy* can be formulated to ensure that two tasks will be performed "simultaneously", i.e., concurrently within a particular unit of time such as a day. (As a result, some tasks may be delayed until other tasks have caught up.)

C3: Explanation of the mitigation strategy(ies)
For example, why a given strategy was identified and what it entails.

Implemented (Y/N): N
Brief description: A mitigation strategy is manually formulated by a clinician in terms of a CIG integration policy; this question seems more suitable for automatically generated mitigation strategies.

That said, the underlying adverse interactions are detected by the MLOD tool, which will show the sources utilized in the detection (e.g., Drugbank, Drugcentral).

C4: Alternative mitigation strategies for a single interaction
For example, if there are more than one possible mitigation strategies, are they identified and presented.

Implemented (Y/N): N
Brief description: This question seems more suitable for automatically generated mitigation strategies. In our case, the MLOD tool will detect underlying adverse interactions, and, in response, the clinician may formulate any type of CIG integration policy based on recommendations from clinical guidelines.

Part 3: Implementation of the Case Studies

Case 1. TIA/Duodenal ulcer/Osteoporosis

Input (1 page):

- Show the encoded CIGs required to solve the case in your approach formalism.

Figures 1-3 show the relevant parts of the CIGs. The summarized code can be found in the `case1/` subfolder.

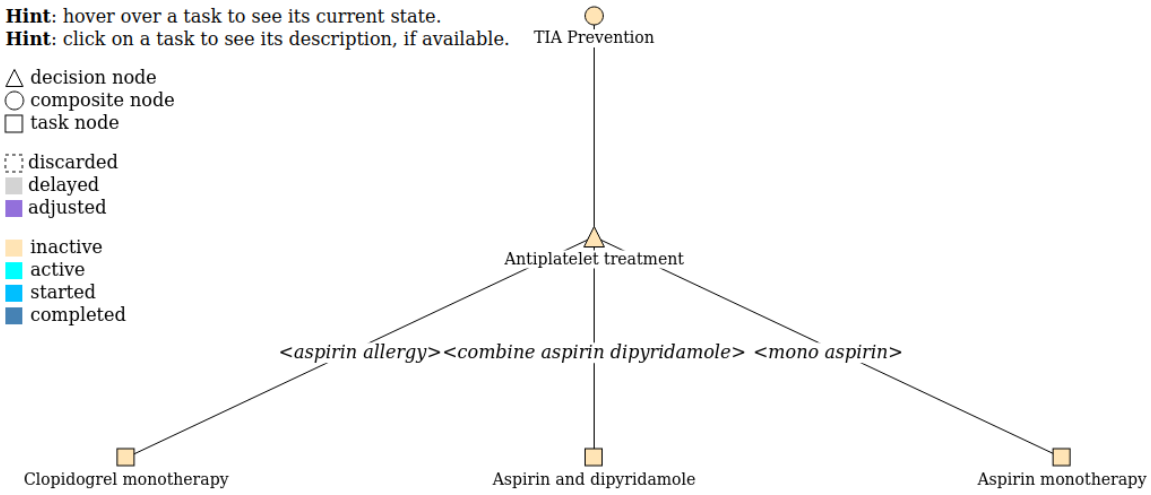


Figure 1. TIA Prevention CIG (relevant part) (online: http://ppr.cs.dal.ca:3005/tia_prevention.html).

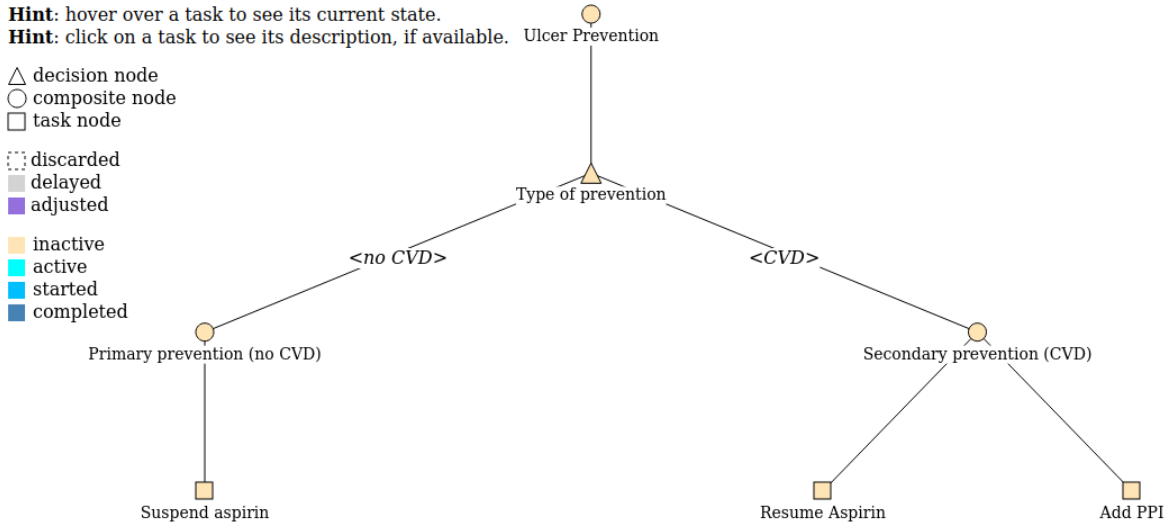


Figure 2. Ulcer Prevention CIG (relevant part) (online: http://ppr.cs.dal.ca:3005/ulcer_prevention.html).

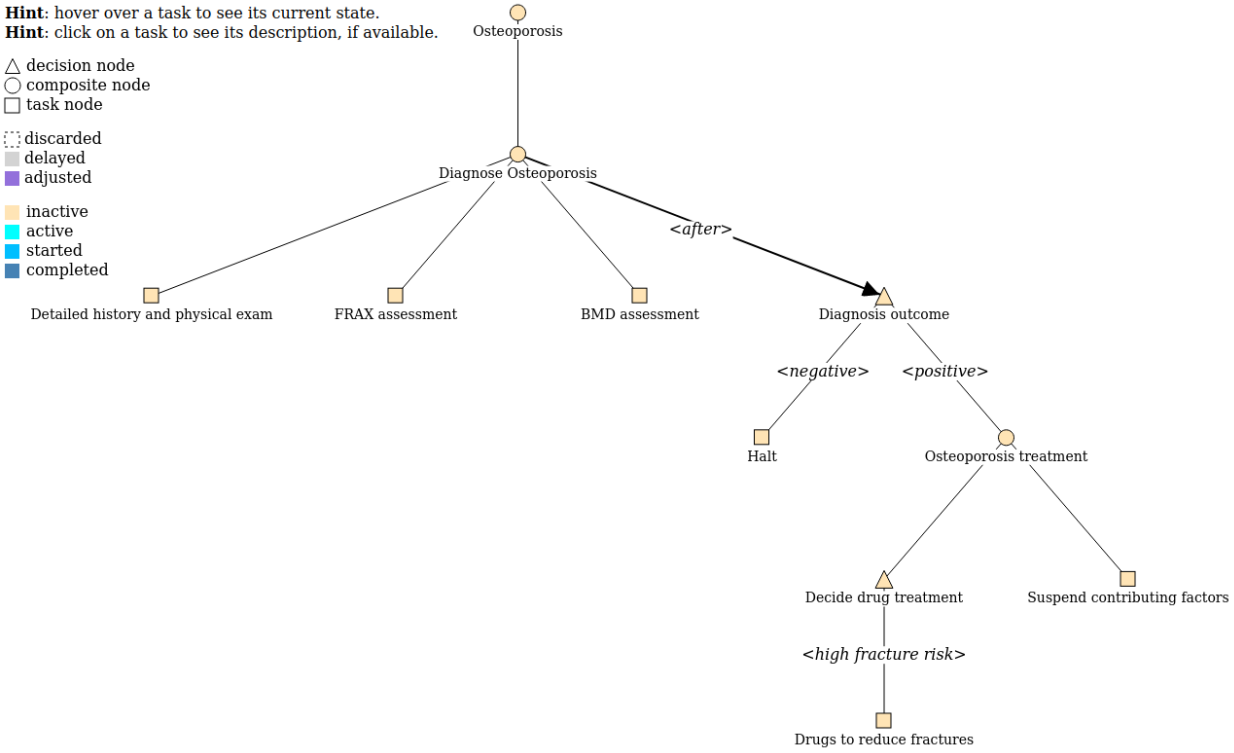


Figure 3. Osteoporosis CIG (relevant part) (online: http://ppr.cs.dal.ca:3005/osteoporosis.html)

- Show the encoded patient data

The patient data is encoded using HL7 FHIR (Turtle syntax) and can be found in `case1/patient1.ttl`.

- If applicable, show how adverse interactions (features A1-A7) were encoded a-priori

N/A

- If applicable, show/reference the encoding of additional domain knowledge

To enable the detection of adverse interactions using the MLOD tool, the relevant CIG tasks are annotated with the drugs being prescribed, using the SNOMED-CT vocabulary. For instance:

```
:aspirin_monotherapy a :CIGTask ;
        rdfs:label "Aspirin monotherapy" ;
        :involves :presc_aspirin .

:presc_aspirin a sct:Prescription;
        sct:drugUsed drug:4244 , drug:DB00945 .

drug:4244 rdfs:label "acetylsalicylic acid" ;
        dc:source "DrugCentral" .
drug:DB00945 rdfs:label "Acetylsalicylic acid"
        dc:source "DrugBank" .
```

Processing (1 page):

- If applicable, explain how relevant interactions were (automatically) identified (features A1-A7)

- **A1**: Drug from a CPG has an effect on a comorbid condition.

The MLOD sources do not contain a drug-condition interaction (DCI) between "acetylsalicylic acid" (aspirin) and "duodenal ulcer disease".

The MLOD sources do contain a drug-condition interaction (DCI) between "Proton pump inhibitors" and "Osteoporosis" in DrugCentral:

```
drug_interaction:78933 a dv:Drug-Condition-Interaction ;
        dv:structure-interactor drug_struct:1055 ; # drug structure of PPI
        dv:condition-interactor drug_condition:10031282 ; # Osteoporosis
        dct:source "DrugCentral" .

drug:7753 dv:drug-structure drug_struct:1055 . # PPI
```

- **A4**: Conflicting actions (e.g., drugs, procedures) from different CPGs.

The MLOD tool detects *potential* interactions between the TIA Prevention and Ulcer Prevention CIG: i.e., several tasks involve the same drugs (i.e., Aspirin: "Aspirin monotherapy", Fig. 1; "Suspend Aspirin", "Resume aspirin", Fig. 2). Similarly, a potential interaction is detected between the Ulcer Prevention and Osteoporosis CIG, as they involve the same drugs (i.e., PPI: "Add PPI", Fig. 2; "Suspend contributing factors", Fig. 3).

These potential interactions will be presented to the clinical expert, who may choose to act on them by formulating a CIG integration policy.

- **A7**: Multiple interactions from different CPGs occurring at the same time.

Coping with multiple interactions is not a problem.

- Explain how relevant interactions were (automatically) mitigated (features B1-B8 [A8-A14+Other mitigation strategies])

- **B1**: Adding a drug to mitigate an adverse effect.

The action of adding PPI, as part of the secondary prevention of Ulcer (Fig. 2), is directly recommended by the Ulcer Prevention CIG, and hence does not require an explicit mitigation.

- **B4**. Replacing a drug with a safer / non-interacting drug / more effective drug for comorbidity.
Due to the interaction between aspirin and duodenal ulcer, the clinician formulates a *Replace Tasks Policy* that replaces "Aspirin monotherapy" with "Clopidogrel monotherapy" (Fig. 1):

```
:Replace_Aspirin_Clopidogrel a :ReplaceTasksPolicy ;
  :taskToReplace cig:aspirin_monotherapy ;
  :replacement cig:clopidogrel_monotherapy .
```

The clinician further formulates a *Discard Tasks Policy* that discards "Resume aspirin" (Fig. 2, right branch):

```
:Discard_Aspirin a :DiscardTasksPolicy ;
  :taskToDiscard cig:resume_aspirin .
```

At execution-time, this task will only be discarded in case the workflow proceeds down this branch.

- **B5**: Discard unsafe/interacting drug.
The clinician formulates a *Discard Tasks Policy* to discard the PPI prescription (Add PPI, Fig. 2, right branch):

```
:Discard_PPI a :DiscardTasksPolicy ;
  :taskToDiscard cig:add_ppi .
```

As before, this task will only be discarded if the workflow proceeds down the right branch.

- If applicable, explain how other relevant features were realized (features C1-C4[A15-A18])

- **C4**: Alternative mitigation strategies for a single interaction.
It is up to the clinician to formulate CIG integration policies to mitigate the interactions that were detected with help from the MLOD tool. A single coherent strategy will need to be devised by the clinician: in this case, involving a *Replace Tasks Policy* and two *Discard Tasks Policy'*s. This strategy meets all goals, namely TIA prevention (clopidogrel) and Ulcer prevention (removing aspirin), and preventing the secondary cause of Osteoporosis (removing PPI).

- Explain which parts of the processing are generic and which need to be hardwired for the case[7]

Processing algorithm: this algorithm is fully generic.

Domain knowledge: As mentioned, the MLOD tool helps the clinician to detect adverse interactions. Subsequently, aided by (a) the MLOD tool output (automated), possibly combined with (b) the CPG content (manual), the clinician manually formulates CIG integration policies to mitigate these issues.

At a future stage, the system could automatically suggest CIG integration policies (e.g., based on drug-drug or drug-condition interactions).

[7] There are two aspects: (**1**) processing algorithm: in a generic approach, only models change across case studies, while a hardwired approach requires tweaking the algorithm for each case study; (**2**) domain knowledge: a mitigation strategy can be generic or hardwired: e.g., deriving which drug should replace another drug can come from a knowledge base or be hard-wired for each case study (e.g., based on guidelines). There can be degrees of generality as well, of course.

Output (1 page):

- Show and explain how the result of the processing is represented

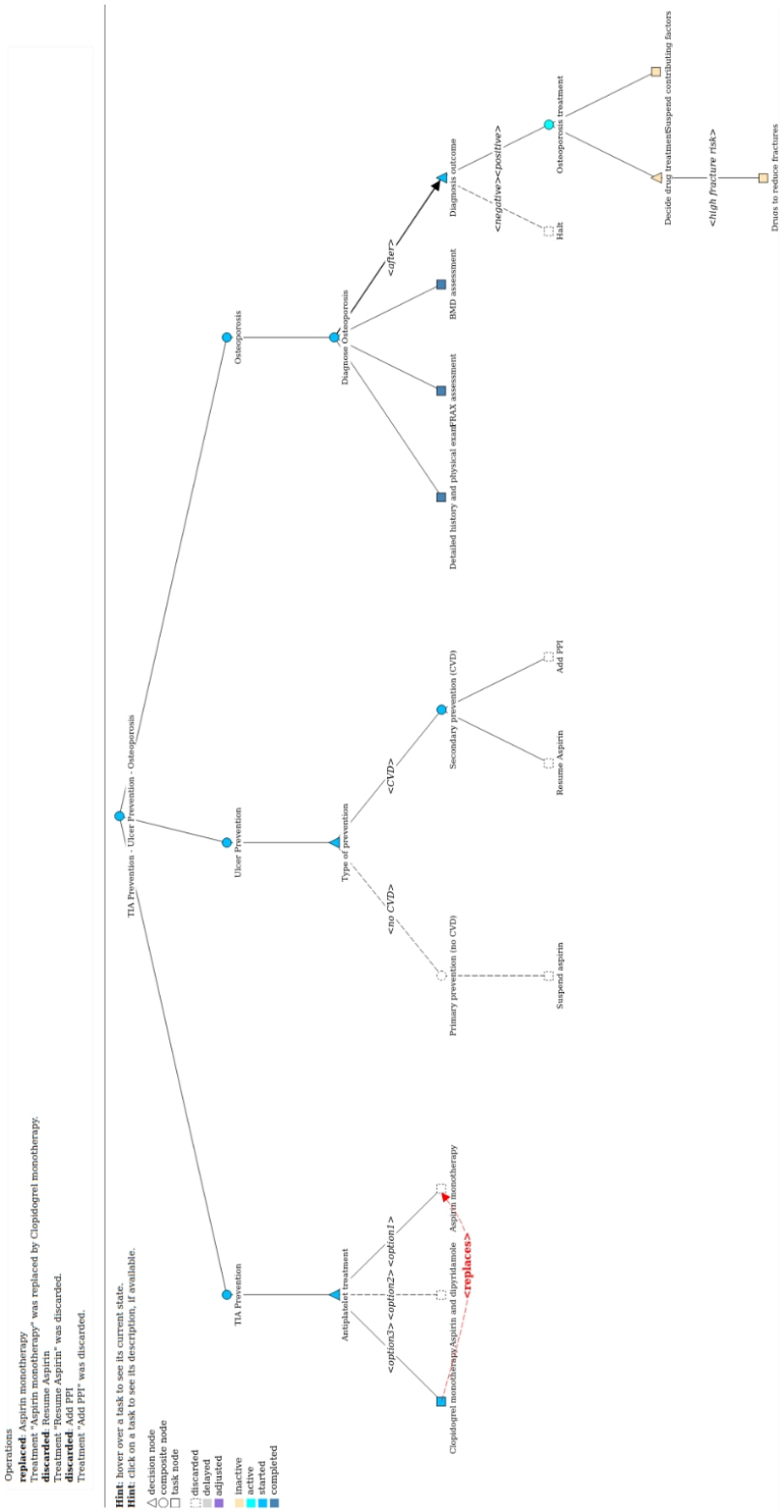


Figure 4. Output of case 1[8] (online: http://ppr.cs.dal.ca:3005/tia_ulcer_osteoporosis.html; click "apply").

[8]In case the labels do not fit on the screen, reduce the zoom of the page, and refresh the browser.

- Show and explain what user interactions were involved in the use case

Analysis of the MLOD tool output and/or CPG contents for adverse interactions, and subsequent formulation of CIG integration policies for mitigation, are performed by the clinician.

- Explain any additional considerations.

N/A

Case 2. Venous Thromboembolism /Urinary tract infection

Input (1 page):

- Show the encoded CIGs required to solve the case in your approach formalism

Figures 4-5 show the relevant parts of the CIGs. The summarized code can be found in the `case3/` subfolder.

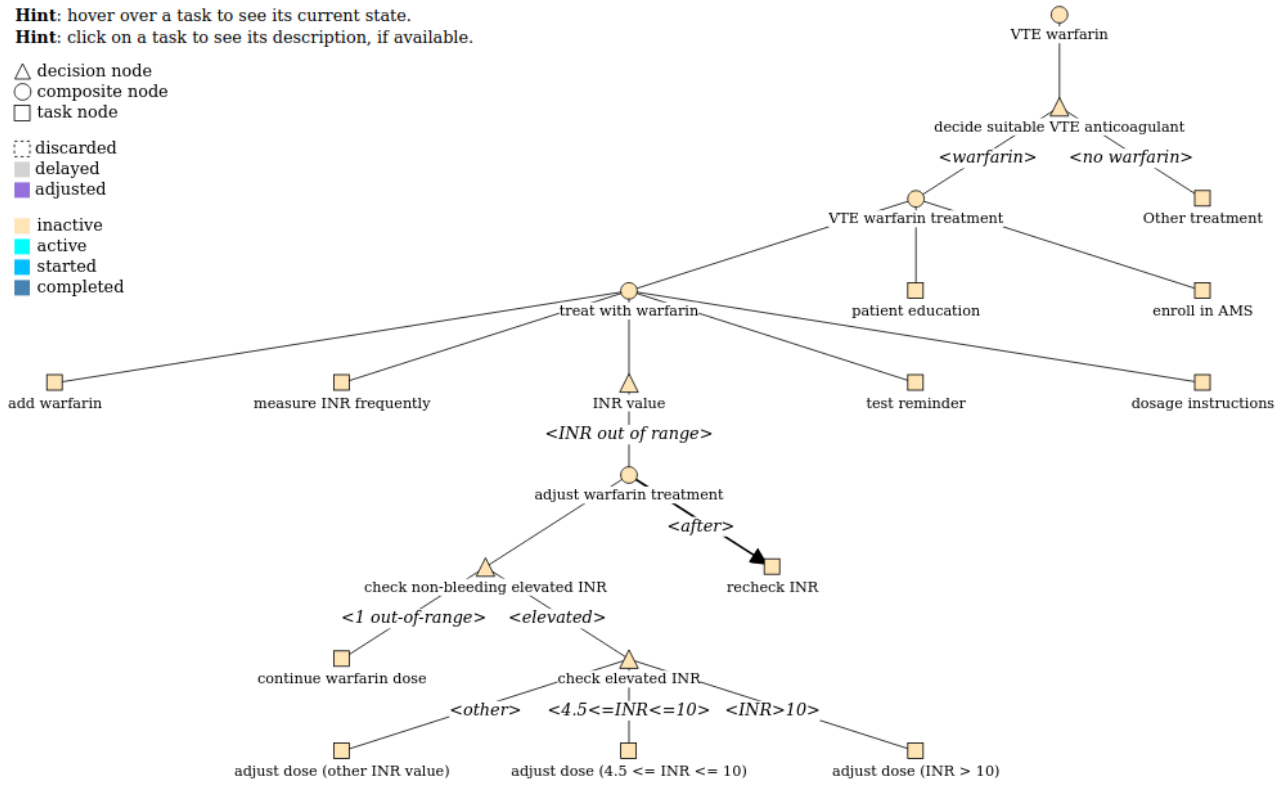


Figure 4. VTE Warfarin treatment (online: http://ppr.cs.dal.ca:3005/vte_warfarin.html).

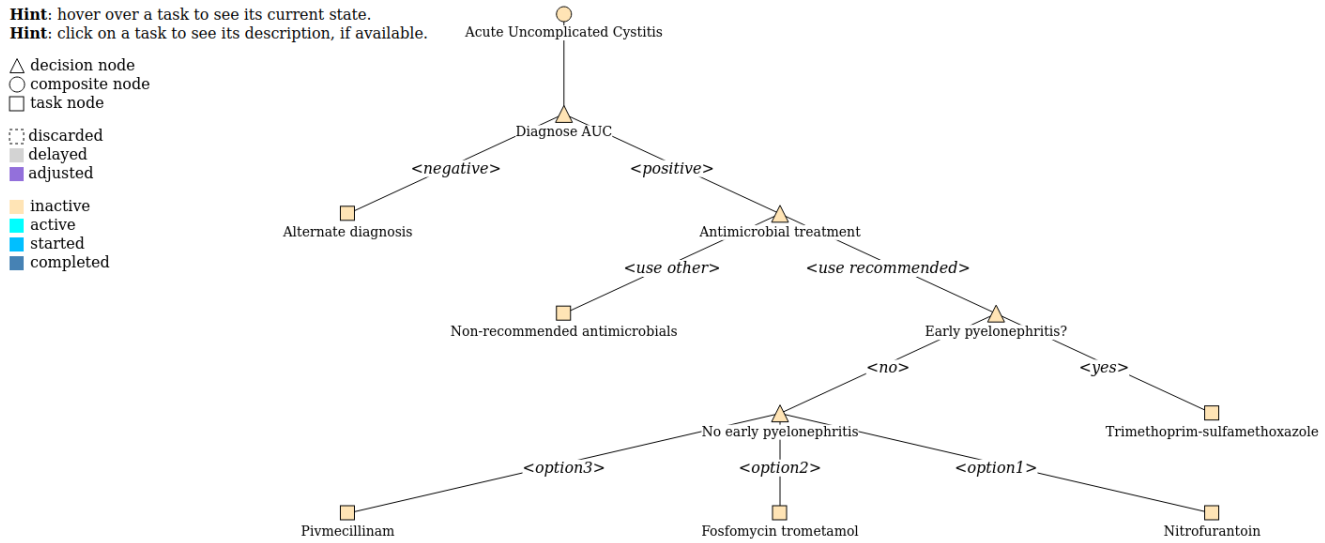


Figure 5. Acute Uncomplicated Cystitis CIG (online: http://ppr.cs.dal.ca:3005/auc.html).

- Show the encoded patient data

The patient data is encoded using HL7 FHIR (Turtle syntax) and can be found in `case3/patient3.ttl`.

- If applicable, show how adverse interactions (features A1-A7) were encoded a-priori

N/A

- If applicable, show/reference the encoding of additional domain knowledge

To enable the detection of adverse interactions using the MLOD tool, the relevant CIG tasks are annotated with the drugs being prescribed, using the SNOMED-CT vocabulary. For instance:

```
:add_warfarin a :CIGTask ;
        :involves :presc_warfarin .
:presc_warfarin a sct:Prescription ;
        sct:drugUsed drug:DB00682 .

drug:DB00682
        rdfs:label "Warfarin" ;
        dc:source "DrugBank" .
```

Processing (1 page):

- If applicable, explain how relevant interactions were (automatically) identified (features A1-A7)

- **A2**: Two or more drugs from different CPGs interact

The MLOD sources find a drug-drug interaction (DDI) between "Warfarin" and "Trimethoprim" in DrugBank:

```
drug_interaction:DB00440_DB00682 a dv:Drug-Drug-Interaction ;
        dv:drug-interactor drug:DB00440 ; # Trimethoprim
        dv:drug-interactor drug:DB00682 ; # Warfarin
        dct:source "DrugBank" ;
        dct:description  "The  metabolism  of  Trimethoprim  can  be  decreased  when  combined  with
Warfarin." .
```

- Explain how relevant interactions were (automatically) mitigated (features B1-B8 [A8-A14+Other mitigation strategies])

- **B2**: Adjust drug dosage
- **B3**: Monitor the effect of a drug

Due to the interaction between Warfarin and Trimethoprim, the clinician formulates an *Event Conditional Replace Policy* that will replace the "add warfarin" task (Fig. 4) with the same prescription but at a reduced 10% dosage, for the duration of the Trimethoprim treatment:

```
:adjust_Warfarin_dose_during_Trimethoprim a :EventCondReplacePolicy ;
        :eventTask cig:treatment_trimethoprim-sulfamethoxazole ; # AUC CIG
        :taskToReplace cig:add_warfarin ; # VTE warfarin CIG
        :replacement [
                :eventTime :during ;
                :task cig:add_warfarin_reduce_10perc
        ] .
```

Further, the clinician formulates an *Event Conditional Replace Policy* that will adjust the monitoring frequency of the INR value during treatment with Trimethoprim. In particular, this policy will replace the "measure INR frequently" with task "measure INR daily" (Fig. 4) for as long as Trimethoprim treatment ensues. (We note that, during that time, in case the INR value becomes out of range, the regular CIG will adjust Warfarin dosage accordingly; see "adjust warfarin treatment" task, Fig. 4.) Once treatment with Trimethoprim has concluded, more frequent INR testing still needs be performed[9] – since the discontinuation of an interacting drug will likely impact INR as well. Once a stable INR has been observed, the increase of INR testing frequency is finally cancelled. This is represented by the following *Event Conditional Replace Policy*:

```
:adjust_INR_freq_during_Trimethoprim a :EventCondReplacePolicy ;
        :eventTask cig:treatment_trimethoprim-sulfamethoxazole ; # AUC CIG
        :taskToReplace cig:measure_inr_frequently ; # VTE warfarin CIG
        :replacement [
                :eventTime :during ;
                :task cig:measure_inr_daily ]
        :replacement [
                :eventTime :after ;
                :task cig:measure_inr_daily ;
                :exitCond [ cig:INR_status cig:stable ] ] . # status determined by subsystem
```

The first replacement will take place while the :*eventTask* is started; the second replacement will take place when the :*eventTask* is completed and while its :*exitCond* is not met (i.e., stable INR).

- If applicable, explain how other relevant features were realized (features C1-C4[A15-A18])

- **C1**: Patient preferences and/or patient burden

It is up to the clinician to manually formulate CIG integration policies that take into account patient preferences and/or patient burden. For instance, in case DOACs are chosen over Warfarin to reduce patient burden, the system would go down a different branch in the relevant CIG (Fig. 4, "no warfarin" option at the top) and this adverse interaction would not take place.

[9] This aspect is not described in the case description; we add it here for completeness.

- Explain which parts of the processing are generic and which need to be hardwired for the case[10]

Processing algorithm: this algorithm is fully generic.

Domain knowledge: As mentioned, the MLOD tool helps the clinician to detect adverse interactions. Subsequently, aided by (a) the MLOD tool output (automated), possibly combined with (b) the CPG content (manual), the clinician manually formulates CIG integration policies to mitigate these issues.

At a future stage, the system could automatically suggest CIG integration policies (e.g., based on drug-drug or drug-condition interactions).

Output (1 page):

- Show and explain how the result of the processing is represented

See Fig. 6:

[10] There are two aspects: (**1**) processing algorithm: in a generic approach, only models change across case studies, while a hardwired approach requires tweaking the algorithm for each case study; (**2**) domain knowledge: a mitigation strategy can be generic or hardwired: e.g., deriving which drug should replace another drug can come from a knowledge base or be hard-wired for each case study (e.g., based on guidelines). There can be degrees of generality as well, of course.

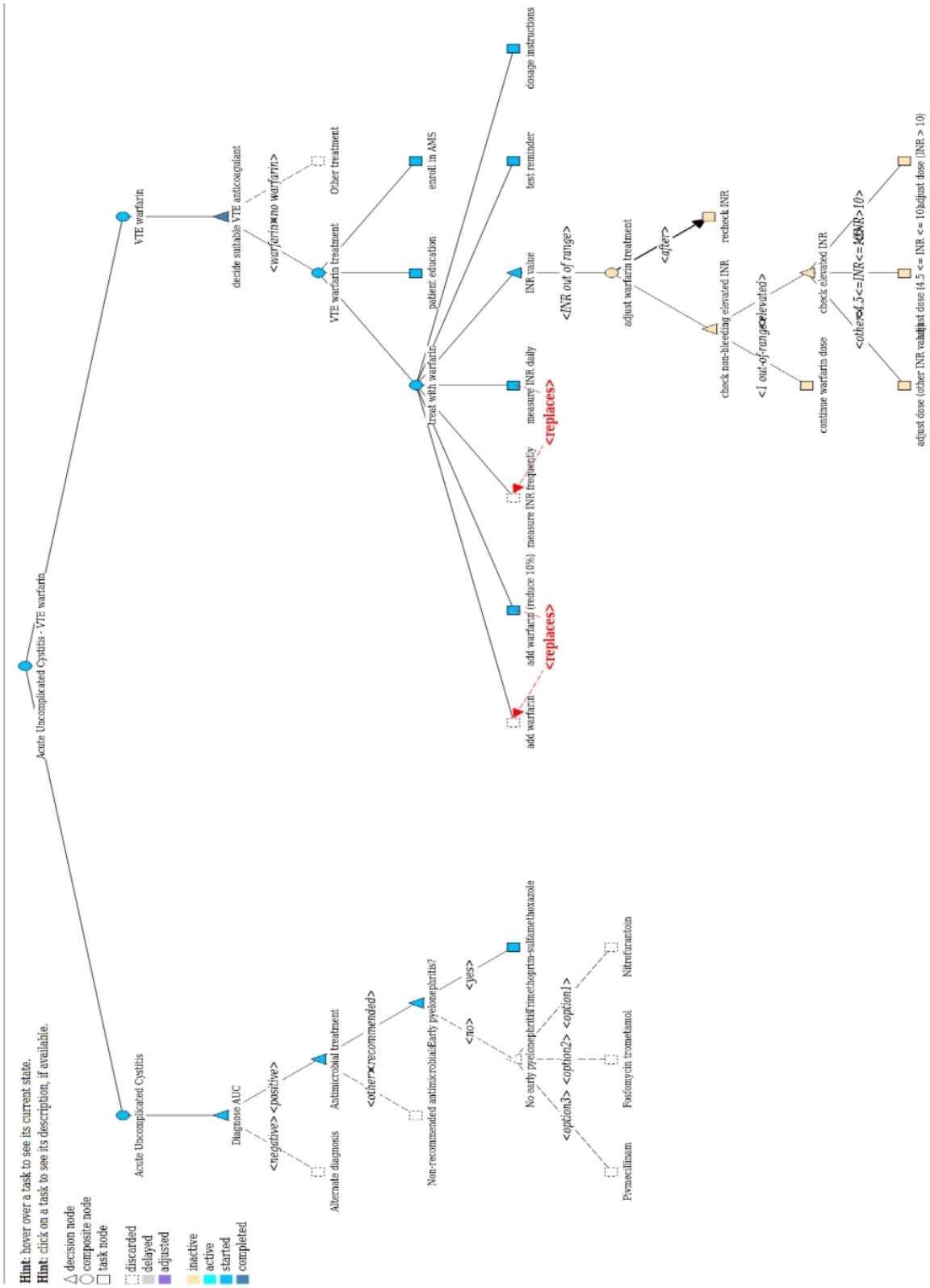


Figure 6. Output of case 3[11] (online: http://ppr.cs.dal.ca:3005/vte_uti.html; select the different steps).

- Show and explain what user interactions were involved in the use case

[11] In case the labels do not fit on the screen, reduce the zoom of the page, and refresh the browser.

Analysis of the MLOD tool output and/or CPG contents for adverse interactions, and subsequent formulation of CIG integration policies for mitigation, are performed by the clinician.

- Explain any additional considerations.

N/A.

References

[1] W. Van Woensel, S. S. R. Abidi, and S. R. Abidi, “Decision support for comorbid conditions via execution-time integration of clinical guidelines using transaction-based semantics and temporal planning,” *Artif. Intell. Med.*, vol. 118, p. 102127, 2021, doi: https://doi.org/10.1016/j.artmed.2021.102127.

[2] W. Van Woensel, “Continuous Transaction Logic: Implementation.” https://github.com/william-vw/jena/tree/master/jena-core (accessed Mar. 31, 2020).

[3] A. J. Bonner and M. Kifer, “An Overview of Transaction Logic,” *Theor. Comput. Sci.*, vol. 133, no. 2, pp. 205–265, 1994, doi: 10.1016/0304-3975(94)90190-2.

[4] B. Jafarpour, S. Raza Abidi, W. Van Woensel, and S. S. Raza Abidi, “Execution-time integration of clinical practice guidelines to provide decision support for comorbid conditions,” *Artif. Intell. Med.*, vol. 94, 2019, doi: 10.1016/j.artmed.2019.02.003.

[5] B. Jafarpour, “Ontology Merging Using Semantically-Defined Merge Criteria and OWL Reasoning Services: Towards Execution-Time Merging of Multiple Clinical Workflows to Handle Comorbidities. PhD Thesis, Dalhousie University, 2013,” Dalhousie University.

[6] W. Van Woensel, S. R. Abidi, B. Jafarpour, and S. S. R. Abidi, “Providing Comorbid Decision Support via the Integration of Clinical Practice Guidelines at Execution-Time by Leveraging Medical Linked Open Datasets,” 2019.

[7] “Medical Linked Open Data (MLOD).” https://github.com/william-vw/mlod (accessed Apr. 28, 2020).

[8] U.S. National Library of Medicine, “SNOMED CT.” https://www.nlm.nih.gov/healthit/snomedct/index.html (accessed Aug. 16, 2019).