

Combining Practical and Dialectical Commitments for Service Engagements

Pankaj R. Telang¹(✉), Anup K. Kalia², John F. Madden³,
and Munindar P. Singh²

¹ Cisco Systems, Research Triangle Park, Durham, NC 27709, USA
ptelang@gmail.com

² North Carolina State University, Raleigh, NC 27695-8206, USA
{akkalia,singh}@ncsu.edu

³ Duke University Medical Center, Durham, NC 27710, USA
john.madden@duke.edu

Abstract. We understand a service engagement as a form of collaboration arising in a sociotechnical system (STS). Although STSs are fruitfully modeled using normative abstractions such as commitments, a conventional (practical) commitment can capture only part of the story, namely, a debtor's promise to the creditor to bring about the consequent if the antecedent holds. In contrast, in a dialectical commitment, which we highlight, a debtor asserts to the creditor that the consequent is true if the antecedent is. For example, a customer may dialectically commit to a seller that the product she received is damaged but may not practically commit to damaging the product. We introduce a novel bipartite operationalization of dialectical commitments that separates their objective and subjective aspects and thus avoids the problems arising if we merely treat dialectical like practical commitments. We express that operationalization in temporal logic, developing a verification tool based on NuSMV, a well-known model-checker, to verify if the participants' interactions comply with the participants' dialectical commitments. We present a set of modeling patterns that incorporate both practical and dialectical commitments. We validate our proposal using a real-world scenario of contradictory medical diagnoses by different specialists.

1 Introduction

A service engagement involves two or more autonomous parties interacting with each other and is thus a prototypical sociotechnical system (STS) [22]. An STS can be fruitfully modeled using normative relationships. To this end, commitments have been extensively employed in modeling service engagements (and associated business processes) [5, 19, 20, 24]. A key benefit of commitments over traditional approaches is that commitments capture outcomes in a declarative manner and minimally constrain the behavior of the participants. Two kinds of commitment are known in the literature [13, 16, 21]: practical and dialectical. In a practical commitment, a debtor agent promises a creditor agent to bring about a condition (consequent) if some other condition (antecedent) holds. For example, a customer may commit to paying a reseller if the reseller delivers the goods.

In a dialectical commitment, the debtor claims that the consequent holds provided the antecedent does. For example, a customer may dialectically commit to a reseller that the customer received the goods in a damaged condition. These commitments differ in the nature of their standard of satisfaction. For example, a customer may dialectically commit that it received damaged goods, but may not practically commit to damaging the goods. Previous research has nearly always considered only practical commitments [6, 25, 26], a recent exception being Baldoni et al. [2].

The present paper incorporates dialectical commitments in modeling an STS to tackle a previously ignored challenge, namely, how participants make claims about putative facts, claims that may be mutually inconsistent. For example, a customer may claim that goods received are damaged whereas the courier may claim the goods delivered were not damaged. Although this paper doesn't tackle norm types other than commitments [22], by bringing forth dialectical commitments, it supports the possibility of modeling disputes between participants in STSs and disparities in their policies. This paper sheds light on how potentially to resolve such disputes, thereby facilitating policy-governed secure collaboration.

Research Question and Contributions. When we model secure collaboration in STSs in normative terms [23], it is important to accommodate disputes among STS participants regarding facts and norms. Previous approaches include the objective, but omit the social, aspect of norms in their lifecycles [22]. This leads to our research question: How can we formalize norms in a manner that incorporates their objective and subjective elements and supports verification of interactions on comprehensive grounds?

This paper is restricted to two norm types: dialectical and practical commitments. It contributes a novel operational model and temporal logic formalization based on Computational Tree Logic (CTL) along with a tool based on NuSMV [17], a CTL model checker, to verify if participants' interactions comply with their commitments. This paper provides a set of modeling patterns incorporating practical and dialectical commitments. We evaluate our approach on a breast cancer diagnosis process specified by a committee of experts called by a major government agency (Office of the Assistant Secretary for Planning and Evaluation (ASPE), US Department of Health and Human Services) [1]. The significance of this work arises from its expanding the operational treatment and formal verification of commitments to incorporate dialectical commitments, thereby enabling new applications that previous approaches cannot tackle.

2 Background

We illustrate the generality of our approach by introducing it via Cisco's Quote to Cash (QTC) business process [25] and evaluating it via a healthcare collaboration scenario (introduced in Sect. 4). The QTC process encompasses all of the key activities that begin from a customer requesting a quote, and end in Cisco receiving payment from the customer. The participants in this process include

customers, resellers, distributors, logistics providers, banks, contract manufacturers, and service providers. A customer purchases goods either directly from Cisco, or from a reseller. In addition to selling the goods, a reseller provides value-added services of installing and configuring goods. A reseller purchases goods either from a distributor, or from Cisco. A distributor always purchases goods from Cisco. Unlike a reseller, a distributor may purchase and stock the goods in its warehouse. To build and ship its products, Cisco uses contract manufacturers and transportation providers respectively. The participants use different banks and credit companies for making payments.

2.1 Practical Commitments

A practical commitment [21] $C(\text{DEBTOR}, \text{CREDITOR}, \text{O-CONTEXT}, \text{antecedent}, \text{consequent})$ means that DEBTOR commits to CREDITOR in the organizational context O-CONTEXT to bring about the consequent provided the antecedent holds. (For brevity, we omit O-CONTEXT where appropriate.) For example, $C(\text{CISCO}, \text{CUSTOMER}, \text{COURT}, \text{pay}, \text{deliver goods})$ means that CISCO commits under the o-context COURT to CUSTOMER to deliver goods, provided CUSTOMER pays.

We describe the lifecycle of a practical commitment from Fig. 1 [25] using the above example. When CISCO creates the commitment, its state changes to **active** from **null**. If CUSTOMER pays CISCO (antecedent holds), the commitment is **detached**. The commitment is **terminated** if CISCO cancels the commitment when **conditional**, or CUSTOMER releases CISCO from the commitment. The commitment is **satisfied** when the goods are delivered (consequent holds). It is **violated** if CUSTOMER has paid up (antecedent holds), but CISCO does not deliver the goods (consequent fails), or if CISCO cancels the commitment. When the commitment is **conditional**, if CUSTOMER does not pay (antecedent fails) CISCO, then the commitment **expires**. If CISCO delegates the commitment to another company (delegatee), CISCO may **suspend** the commitment, making it **pending**. If the delegatee company fails to provide goods to CUSTOMER, then CISCO may **reactivate** its commitment to CUSTOMER.

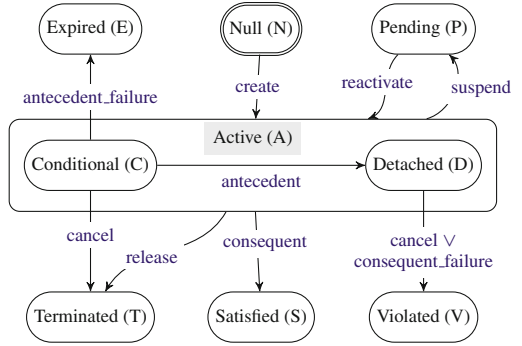


Fig. 1. Practical commitment lifecycle as a state transition diagram.

2.2 Computation Tree Logic

Computation Tree Logic (CTL) [4] is a temporal logic based on a branching time structure. Each temporal operator in CTL has two components. The first

component is a path quantifier: either A , meaning on all of the paths; or E , meaning on at least one path. The second component is a linear-time operator: F , meaning in a future state; G , meaning (globally) in all future states; and X , meaning in the next state. A CTL formula may contain the standard logical operators: \neg , \wedge , \vee , and \rightarrow , meaning negation, conjunction, disjunction, and implication, respectively. As an example, $AG(p \rightarrow AFq)$ means that on all paths if proposition p holds in a state, then on all paths emanating from that state proposition q holds in a future state.

3 Dialectical Commitments

The lifecycle of a practical commitment, as shown in Fig. 1, is inadequate for capturing the semantics of a dialectical commitment. For example, consider Fig. 2, which shows a possible execution in which CUSTOMER and CISCO interact to decide if the goods delivered to CUSTOMER are damaged. CUSTOMER informs CISCO, i.e., dialectically commits that the goods are damaged.

However, CISCO disagrees with CUSTOMER and challenges CUSTOMER's claim. That is CISCO dialectically commits that the goods are not damaged. CUSTOMER then requests a relevant higher authority, such as COURT, for resolution. COURT concludes that the goods are not damaged. CUSTOMER agrees with COURT and retracts its dialectical commitment.

If we employ the lifecycle of a practical commitment to handle CUSTOMER's dialectical commitment, then the commitment is violated since COURT concludes that the goods are not damaged. Thus, the lifecycle of a practical commitment fails to handle CUSTOMER's retraction of a dialectical commitment appropriately.

We write a dialectical commitment using a notation similar to that of a practical commitment: $D(\text{DEBTOR}, \text{CREDITOR}, \text{O-CONTEXT}, \text{antecedent}, \text{consequent})$. For example, $D(\text{CUSTOMER}, \text{CISCO}, \text{COURT}, \top, \text{goods-damaged})$ means that CUSTOMER dialectically and unconditionally (antecedent is \top , true) commits to CISCO that the goods are damaged. We allow the debtor to be a set of roles. For example, in the QTC process, CUSTOMER and RESELLER may jointly commit to CISCO that the goods are damaged: $D(\{\text{CUSTOMER}, \text{RESELLER}\}, \text{CISCO}, \text{COURT}, \top, \text{goods-damaged})$.

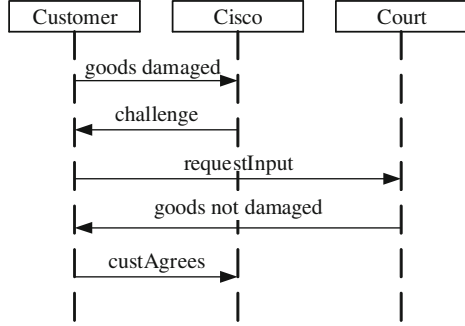


Fig. 2. Customer and Cisco interactions.

3.1 The Proposed Lifecycle of Dialectical Commitments

Figure 3 shows our proposed lifecycle of a dialectical commitment. The state of a dialectical commitment has two dimensions: objective (computed based on

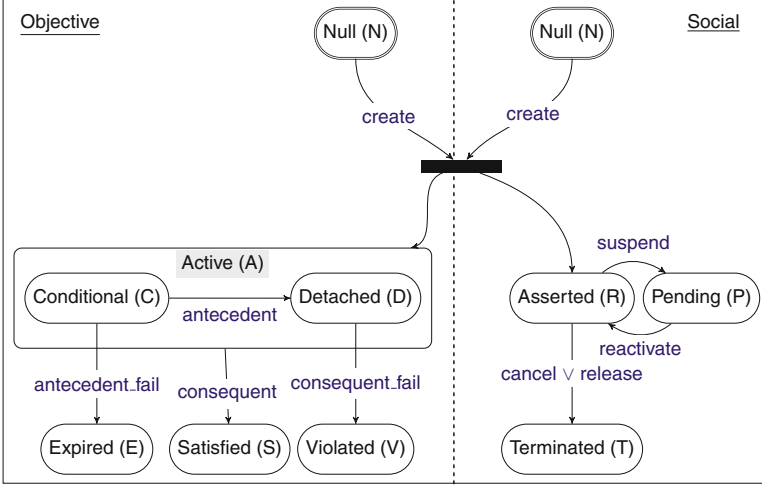


Fig. 3. Dialectical commitment lifecycle as a state-transition diagram.

the antecedent and consequent, treated as objective facts) and social (computed based on the creditor or the debtor's actions). Thus, the state of a dialectical commitment is bipartite and written as a pair, e.g., $\langle \textit{satisfied}, \textit{asserted} \rangle$.

A dialectical commitment is **null** before it is created. Upon creation, its objective state becomes **active** and its social state becomes **asserted**. The active state has two substates: **conditional** and **detached**. The commitment becomes **detached** when its antecedent holds. If the antecedent of a **conditional** commitment fails, then the commitment becomes **expired**. The commitment is **satisfied** if its consequent becomes true when it is **active**. The commitment becomes **violated** if its consequent fails when it is **detached**. On the social side, if the debtor cancels or suspends the commitment when it is **asserted**, it becomes **terminated** or **pending**, respectively. If the debtor reactivates the commitment when it is **pending**, it becomes **asserted**. We write the bipartite state of a dialectical commitment as a pair: $\langle \textit{objective-state}, \textit{social-state} \rangle$. We write the objective state as $D_{\text{obj}}^{\textit{ostate}}$ and social state as $D_{\text{soc}}^{\textit{sstate}}$, where $\textit{ostate} \in \{N, C, D, E, S, V\}$ and $\textit{sstate} \in \{N, R, P, T\}$ (state labels are from Fig. 3).

We describe the progression of a dialectical commitment in the CUSTOMER-CISCO interactions from Fig. 2. CUSTOMER informs CISCO that the goods are damaged and thus creates the dialectical commitment: $D_u = D(\text{CUSTOMER}, \text{CISCO}, \text{COURT}, \top, \text{goods-damaged})$. Upon creation, D_u 's state is $\langle \textit{detached}, \textit{asserted} \rangle$ (*detached* since its antecedent is true (\top)).

However, CISCO disagrees with CUSTOMER and challenges CUSTOMER's claim, thus creating the dialectical commitment: $D_c = D(\text{CISCO}, \text{CUSTOMER}, \text{COURT}, \top, \neg\text{goods-damaged})$. Upon creation, D_c is $\langle \textit{detached}, \textit{asserted} \rangle$. CUSTOMER and CISCO may resolve their difference of opinion among themselves. But in Fig. 2 they escalate the dispute to COURT on the condition of goods. COURT concludes that the goods are not damaged, which causes D_u to transition to $\langle \textit{violated},$

asserted), and D_c to transition to $\langle \textit{satisfied}, \textit{asserted} \rangle$. Finally, CUSTOMER agrees that the goods are not damaged, that is, CUSTOMER cancels D_u , causing its state to transition to $\langle \textit{violated}, \textit{terminated} \rangle$.

3.2 Formalization

We now formalize the lifecycle of dialectical commitments in CTL. We group the specifications into four groups: state-action, state-state, terminal states, and acceptable executions. To save the space, we describe one from each group.

State-Action Transitions. The CTL specifications for state-action transitions follow from the lifecycle given above. For brevity, we explain only a few of them in English.

- SA1. $AG (D_{obj}^N \wedge \textit{create} \wedge \neg \textit{antecedent} \rightarrow AX D_{obj}^C)$
- SA2. $AG (D_{obj}^N \wedge \textit{create} \wedge \textit{antecedent} \rightarrow AX D_{obj}^D)$
- SA3. $AG (D_{obj}^C \wedge \textit{antecedent} \rightarrow AX D_{obj}^D)$
- SA4. $AG (D_{obj}^C \wedge \textit{antecedent_fail} \rightarrow AX D_{obj}^E)$
- SA5. $AG (D_{obj}^D \wedge \textit{consequent_fail} \rightarrow AX D_{obj}^V)$
- SA6. $AG (D_{obj}^{C \vee D} \wedge \textit{consequent} \rightarrow AX D_{obj}^S)$

On any path, if a dialectical commitment is **conditional** or **DETACHED** in a state and its consequent holds, then on all paths emanating from that state in the next state, the commitment's objective state becomes **satisfied**.

- SA7. $AG (D_{soc}^N \wedge \textit{create} \rightarrow AX D_{soc}^R)$

On any path, if a dialectical commitment's social state is **null** in a state and the debtor creates it, then on all paths emanating from that state in the next state, the commitment's social state becomes **asserted**.

- SA8. $AG (D_{soc}^R \wedge \textit{suspend} \rightarrow AX D_{soc}^P)$
- SA9. $AG (D_{soc}^P \wedge \textit{reactivate} \rightarrow AX D_{soc}^R)$
- SA10. $AG (D_{soc}^R \wedge (\textit{cancel} \vee \textit{release}) \rightarrow AX D_{soc}^T)$

SA1 means that on any path, if a dialectical commitment is **null** in a state, the antecedent is not holding, and the debtor creates it, then on all paths emanating from that state, the commitment objectively becomes **conditional** in the next state.

State-State Transitions. These follow from the dialectical commitment lifecycle.

- SS1. $AG (D_{obj}^N \rightarrow AX D_{obj}^{N \vee C \vee E \vee D \vee S \vee V})$
- SS2. $AG (D_{obj}^C \rightarrow AX D_{obj}^{C \vee E \vee D \vee S})$
- SS3. $AG (D_{obj}^D \rightarrow AX D_{obj}^{D \vee V \vee S})$
- SS4. $AG (D_{soc}^N \rightarrow AX D_{soc}^{N \vee R})$
- SS5. $AG (D_{soc}^R \rightarrow AX D_{soc}^{R \vee T \vee P})$
- SS6. $AG (D_{soc}^P \rightarrow AX D_{soc}^{P \vee R})$

SS1 means if a dialectical commitment is objectively **null** in a state, then on all paths emanating from that state, in the next state, the commitment may objectively remain **null** or may transition to **conditional**, **expired**, **detached**, **satisfied**, or **violated**.

Terminal States. These follow from the dialectical commitment lifecycle.

$$\begin{aligned} \text{TS1. } & \text{AG } (D_{obj}^E \rightarrow \text{AX } D_{obj}^E) \\ \text{TS3. } & \text{AG } (D_{obj}^V \rightarrow \text{AX } D_{obj}^V) \end{aligned}$$

$$\begin{aligned} \text{TS2. } & \text{AG } (D_{obj}^S \rightarrow \text{AX } D_{obj}^S) \\ \text{TS4. } & \text{AG } (D_{soc}^T \rightarrow \text{AX } D_{soc}^T) \end{aligned}$$

TS1 means on any path, if a dialectical commitment is objectively expired in a state, then on all paths emanating from that state in the next state, the commitment objectively remains expired.

Acceptable Executions. The above CTL specifications, which follow from the lifecycle, represent hard integrity requirements on the executions. The participants may have additional requirements on acceptable executions. We now describe some common acceptable executions.

$$\begin{aligned} \text{AE1. } & \text{AF AG } (D_{obj}^N \vee D_{obj}^C) \\ \text{AE3. } & \text{AF AG } (D_{obj}^S \wedge D_{soc}^R) \end{aligned}$$

$$\begin{aligned} \text{AE2. } & \text{AF AG } (D_{obj}^E \wedge D_{soc}^R) \\ \text{AE4. } & \text{AF AG } (D_{obj}^V \wedge D_{soc}^T) \end{aligned}$$

AE1 means an execution is acceptable if a dialectical commitment is never created or remains forever conditional on it. AE2 means an execution is acceptable if a dialectical commitment is created but later expires. AE3 means on an execution, a dialectical commitment may be objectively satisfied and socially asserted, i.e., $\langle \text{satisfied}, \text{asserted} \rangle$. However, such an execution may be acceptable since the debtor is asserting a statement that is deemed objectively true. AE4 means on an execution, a debtor may create a dialectical commitment whose consequent turns out to be false, that is, the commitment transitions to: $\langle \text{violated}, \text{asserted} \rangle$. In such a case, the debtor should cancel the commitment thus transitioning its state to: $\langle \text{violated}, \text{terminated} \rangle$. Debtor's cancellation implies that the debtor acknowledges its error. In some scenarios, debtor may be penalized for such fallacies—the context may create a commitment in which the debtor is required to pay a penalty to the creditor.

The CTL specification capturing the above desirable states of a dialectical commitment is: $\text{AF AG } (D_{obj}^N \vee D_{obj}^C \vee (D_{obj}^S \wedge D_{soc}^R) \vee (D_{obj}^V \wedge D_{soc}^T))$. This specification means that on all paths in the future, a dialectical commitment's objective state remains *null* or *conditional*, or its objective and social state becomes $\langle \text{satisfied}, \text{asserted} \rangle$, or $\langle \text{violated}, \text{terminated} \rangle$.

These examples pertain to executions ending up in certain states. In some cases, the participants may desire executions that pass through some intermediate states. We can state and verify additional properties on intermediate states as well. For example, we can write the requirement that D should always be created as: $\text{AF } D_{obj}^{C \vee D}$.

3.3 Modeling Patterns

This section presents a nonexhaustive set of representative modeling patterns.

Service Provisioning with Claimed Correctness. A provider (1) practically commits to a client to bring about a consequent condition if some antecedent condition holds, and (2) dialectically commits that either the client would agree

with the consequent, or in case of a disagreement between the client and the provider, a higher authority would agree with the consequent.

$$\begin{aligned} C_1 &= C(\text{PROVIDER}, \text{CLIENT}, \text{ant}, \text{con}) \\ D_1 &= D(\text{PROVIDER}, \text{CLIENT}, \text{con}, \text{clientAgrees} \vee \text{authAgrees}) \end{aligned}$$

For example, RESELLER (practically) commits to CUSTOMER to providing and installing the goods if CUSTOMER pays: $C(\text{RESELLER}, \text{CUSTOMER}, \text{pay}, \text{goods} \wedge \text{INSTALL})$. And RESELLER dialectically commits to CUSTOMER that the goods will be in a working condition, and the installation service acceptable: $D(\text{RESELLER}, \text{CUSTOMER}, \text{goods}, \text{clientGoodsWorking} \vee \text{authGoodsWorking})$, $D(\text{RESELLER}, \text{CUSTOMER}, \text{install}, \text{clientAcceptableInstallation} \vee \text{authAcceptableInstallation})$.

Escalation. O-CONTEXT commits to bringing about the creation of a commitment (C_2) that if the PROVIDER violates its commitment (C_1), and the CLIENT escalates the (presumed) violation to the O-CONTEXT. In C_3 , another PROVIDER commits to CLIENT to bring about the consequent. Additionally, the O-CONTEXT may penalize the violating PROVIDER or not, depending on the modeled settings and the particular circumstances that obtain, some of which need not concern CLIENT.

$$\begin{aligned} C_1 &= C(\text{PROVIDER}, \text{CLIENT}, \text{ant}, \text{con}) \\ C_2 &= C(\text{o-context}, \text{CLIENT}, \text{vio}(C_1) \wedge \text{escalate}, \text{create}(C_3)) \\ C_3 &= C(\text{PROVIDER}', \text{CLIENT}, \text{ant}', \text{con}) \end{aligned}$$

For example, DISTRIBUTOR practically commits to delivering goods to CUSTOMER: $C_1 = C(\text{DISTRIBUTOR}, \text{CUSTOMER}, \top, \text{goods})$. If DISTRIBUTOR fails to deliver the goods, the context COURT directs another distributor to deliver the goods: $C_2 = C(\text{COURT}, \text{CUSTOMER}, \text{vio}(C_1) \wedge \text{escalate}, \text{create}(C_3))$, $C_3 = C(\text{DISTRIBUTOR}', \text{CUSTOMER}, \top, \text{goods})$.

Chained Service Provisioning with Jointly Claimed Correctness. Provider SP1 commits to a client to bring about a consequent if some antecedent holds. Additionally, SP1 dialectically commits that either the client or (in case of a disagreement between the client and the provider) a higher authority would agree that the consequent holds, if providers SP2 and SP3 do not violate their dialectical commitment (D_2). SP2 and SP3 jointly dialectically commit to SP1 that either SP1 or (in case of a disagreement) a higher authority would agree that con-3 holds.

$$\begin{aligned} C_1 &= C(\text{SP1}, \text{CLIENT}, \text{ant1}, \text{con1}) \\ D_1 &= D(\text{SP1}, \text{CLIENT}, \neg \text{vio}(D_2) \wedge \text{con1}, \text{clientAgreeCon1} \vee \text{authAgreeCon1}) \\ C_2 &= C(\text{SP3}, \text{SP2}, \text{ant2}, \text{con2}) \\ C_3 &= C(\text{SP2}, \text{SP1}, \text{ant3}, \text{con3}) \\ D_2 &= C(\{\text{SP2}, \text{SP3}\}, \text{SP1}, \text{con3}, \text{sp1AgreeCon3} \vee \text{authAgreeCon3}) \end{aligned}$$

4 Evaluation

We evaluate our approach on a breast cancer diagnosis process specified by a committee of experts called by a major government agency (US Department of Health and Human Services) [1]. This process models five roles: PATIENT, PHYSICIAN, RADIOLOGIST, PATHOLOGIST, and REGISTRAR. The roles interact as follows: (1) the physician orders a mammography (imaging) exam for the patient; (2) if the radiologist notices suspicious calcifications, she recommends a biopsy; (3) if the physician agrees, she performs a biopsy, and sends the collected tissue specimen to the pathologist; (4) the pathologist analyzes the specimen, and performs ancillary studies; (5) the pathologist and radiologist may confer to reconcile their results and produce a consensus report; (6) the physician reviews the integrated report with the patient to create a treatment plan; and (7) the pathologist forwards his report to a cancer registry's registrar.

We apply the patterns on the cancer diagnosis scenario to produce a commitment-based model. We rename the pattern roles with the scenario-specific role names, and substitute the scenario-specific tasks as the antecedents and consequents of the appropriate commitments. We describe the commitments shown in Table 1 and the patterns that compose the model.

Table 1. Commitment-based model for the diagnosis process

C ₁	C(PHY, PAT, diagReq \wedge \neg vio(C ₂) \wedge \neg vio(C ₃), diag)
C ₂	C(PAT, PHY, iApptReq, iApptKept)
C ₃	C(PAT, PHY, bApptReq, bApptKept)
C ₄	C(RAD, PHY, biopsyReq \wedge bApptKept, radPathResults)
C ₅	C(RAD, PHY, imagingReq \wedge iApptKept, imagingResults)
C ₆	C(PATH, RAD, pathologyReq \wedge tissue, pathResults)
C ₇	C(PATH, HOSP, patHasCancer, patRepToRegistrar)
C ₈	C(REG, HOSP, patRepToRegistrar, addPatToRegistry)
C ₉	C(HOSP, PHY, vio(C ₅) \wedge esc, create(C ₅ ') \wedge create(D ₂ '))
C ₁₀	C(BOARD, RAD, radReq, BAgreesPath \vee BDisagreesPath)
C ₁₁	C(BOARD, PHY, phyReq, BAgreesRad \vee BDisagreesRad)
C ₁₂	C(BOARD, PAT, patReq, BAgreesPhy \vee BDisagreesPhy)
D ₁	D(PHY, PAT, diag \wedge \neg vio(D ₃), patAgrees \vee BAgreesDiag)
D ₂	D(RAD, PHY, imaging, phyAgreesI \vee BAgreesI)
D ₃	D({RAD, PATH}, PHY, radPathResults, phyAgreesRP \vee BAgreesRP)
D ₄	D(RAD, PATH, radResults, pathAgreesR \vee BAgreesR)
D ₅	D(PATH, RAD, pathResults, radAgreesP \vee BAgreesP)

(PHY: PHYSICIAN, PAT: PATIENT, RAD: RADIOLOGIST, BOARD: TUMOR BOARD, REG: REGISTRAR, HOSP: HOSPITAL)

Patient's Appointments. Practical commitments (C_2, C_3). PATIENT commits to PHYSICIAN to keep her imaging (C_2) and biopsy appointments (C_3) if requested.

Add Patient to Registry. Practical commitments (C_7, C_8). PATHOLOGIST commits to HOSPITAL (C_7) to reporting PATIENT to REGISTRAR if PATIENT has cancer, and REGISTRAR commits to HOSPITAL (C_8) to add PATIENT to the registry.

Patient's Radiology and Pathologist's Diagnosis. Chained service provider with jointly claimed correctness (C_1, D_1, C_4, C_6, D_3). PATHOLOGIST commits to RADIOLOGIST (C_6) to provide a pathology report if RADIOLOGIST requests it and provides a tissue sample. RADIOLOGIST commits to PHYSICIAN (C_4) to provide an integrated radiology and pathology report if PHYSICIAN requests it and PATIENT keeps the necessary appointment. PATHOLOGIST and RADIOLOGIST jointly dialectically commit to PHYSICIAN (D_3) regarding the correctness of the integrated report. PHYSICIAN commits to PATIENT (C_1) to provide a diagnosis report if PATIENT requests it and keeps necessary appointments. PHYSICIAN dialectically commits to PATIENT (D_1) to the correctness of the diagnosis report if the integrated radiology and pathology report is correct.

Patient's Imaging. Service provisioning with correctness (C_5, D_2). RADIOLOGIST commits to PHYSICIAN (C_5) to provide imaging results if PHYSICIAN requests the results. In addition, RADIOLOGIST dialectically commits to PHYSICIAN (D_2) regarding the correctness of the imaging results.

Escalate Radiologist's Failure to Provide Imaging Results. Escalate (C_5, C_9, C_5', D_2'). HOSPITAL commits to PHYSICIAN to bring about the creation of practical (C_5') and dialectical (D_2') commitments from an alternative RADIOLOGIST if the original RADIOLOGIST violates commitment C_5 and PHYSICIAN escalates the violation.

Tumor Board Provides Input on a Diagnosis. Practical commitments ($C_{10}, C_{11}, C_{12}, C_{13}$). TUMOR BOARD commits to PHYSICIAN, RADIOLOGIST, PATIENT, and PATHOLOGIST to provide its input on a diagnosis upon request.

Radiologist and Pathologist Guarantee their Diagnoses. Dialectical commitments (D_4, D_5). RADIOLOGIST dialectically commits (D_4) to PATHOLOGIST that upon providing the radiology report, either PATHOLOGIST would agree with those results, or in the case of a disagreement, TUMOR BOARD will agree with those results. PATHOLOGIST makes a similar commitment (D_5) to RADIOLOGIST regarding the pathology report.

4.1 Verification

This section applies our verification approach to the ASPE process. We adopt the UML 2.0 Sequence Diagram notation [18] to create sequence diagrams for the model from Table 1. Figure 4 shows one of the sequence diagrams. The condition on the outer opt(ional) block is that RADIOLOGIST has reported the imaging

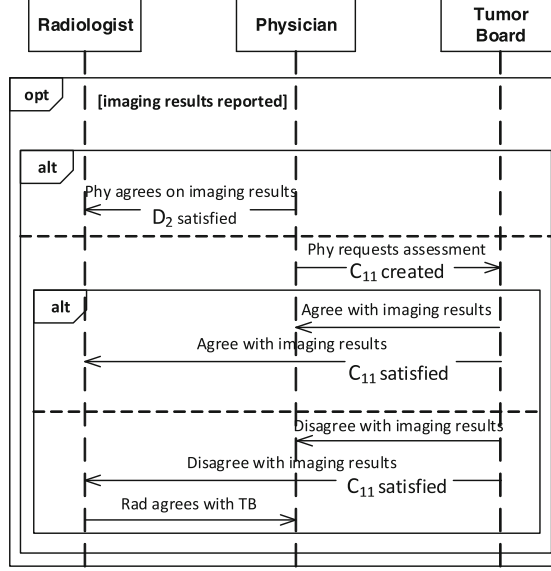
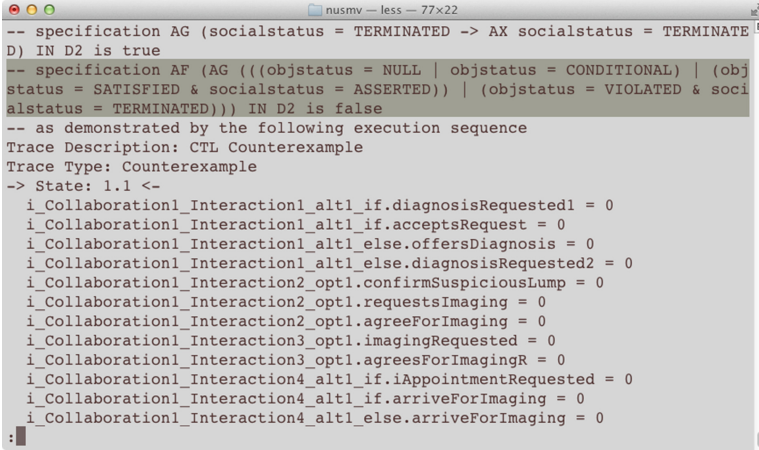


Fig. 4. PHYSICIAN requests TUMOR BOARD to review the imaging results.

results (whether PATIENT has cancer or not) to PHYSICIAN, and created D_2 . In the nested alt(ernate) block, PHYSICIAN either agrees with the imaging results, thus satisfying D_2 , or requests TUMOR BOARD for an assessment, thus creating C_{11} . In the inner alt(ernate) block, TUMOR BOARD either agrees or disagrees with the imaging results. In either case, TUMOR BOARD satisfies C_{11} . If TUMOR BOARD disagrees with the imaging results, RADIOLOGIST cancels and retracts D_2 by informing PHYSICIAN her agreement with TUMOR BOARD.

We develop a NuSMV module for dialectical commitments. We employ this module in verifying models that contain dialectical commitments. Our verification tool (based on NuSMV) [10] takes sequence diagrams and a commitment model as the input. It reports if the sequence diagrams comply with the commitments in the model.

On a computer with 2.66 GHz Intel Core 2 Duo processor, and 8 GB memory, our tool verified the set of sequence diagrams we developed for this scenario (including the one from Fig. 4) in 0.2s. Our tool reported that the sequence diagrams satisfy the model from Table 1. To demonstrate how our approach detects an error, we remove the message from RADIOLOGIST to PHYSICIAN agreeing to TUMOR BOARD’s assessment from the sequence diagram in Fig. 4. Figure 5 shows a partial screenshot of the NuSMV output demonstrating that the model fails to satisfy the (highlighted) CTL specification. The specification shows that $AF\ AG\ (D_{obj}^N \vee D_{obj}^C \vee (D_{obj}^S \wedge D_{soc}^R) \vee (D_{obj}^V \wedge D_{soc}^T))$ is false for D_2 . The counterexample shows a trace in which RADIOLOGIST violates D_2 ; i.e., D_2 remains in the state (violated, asserted). This means RADIOLOGIST does not agree with TUMOR BOARD’s recommendation, and does not cancel D_2 .



```

-- specification AG (socialstatus = TERMINATED -> AX socialstatus = TERMINATE
D) IN D2 is true
-- specification AF (AG (((objstatus = NULL | objstatus = CONDITIONAL) | (obj
status = SATISFIED & socialstatus = ASSERTED)) | (objstatus = VIOLATED & soci
alstatus = TERMINATED))) IN D2 is false
-- as demonstrated by the following execution sequence
Trace Description: CTL Counterexample
Trace Type: Counterexample
-> State: 1.1 <-
  i_Collaboration1_Interaction1_alt1_if.diagnosisRequested1 = 0
  i_Collaboration1_Interaction1_alt1_if.acceptsRequest = 0
  i_Collaboration1_Interaction1_alt1_else.offersDiagnosis = 0
  i_Collaboration1_Interaction1_alt1_else.diagnosisRequested2 = 0
  i_Collaboration1_Interaction2_opt1.confirmSuspiciousLump = 0
  i_Collaboration1_Interaction2_opt1.requestsImaging = 0
  i_Collaboration1_Interaction2_opt1.agreeForImaging = 0
  i_Collaboration1_Interaction3_opt1.imagingRequested = 0
  i_Collaboration1_Interaction3_opt1.agreesForImagingR = 0
  i_Collaboration1_Interaction4_alt1_if.iAppointmentRequested = 0
  i_Collaboration1_Interaction4_alt1_if.arriveForImaging = 0
  i_Collaboration1_Interaction4_alt1_else.arriveForImaging = 0

```

Fig. 5. Tool output indicating an error in the sequence diagrams with respect to the commitments.

4.2 Benefits of Dialectical Commitments

Our approach captures relationships between the participants in terms of practical and dialectical commitments and omits the internal activities of individual participants (e.g., PATHOLOGIST’s slides activity). In this way, it avoids tight coupling between the participants. In addition, our approach provides a basis for answering some significant questions, which the traditional approach cannot answer.

What happens if the treatment plan turns out to be incorrect? Who is or are accountable? An incorrect treatment plan arises from an incorrect integrated radiology and pathology report, which means RADIOLOGIST and PATHOLOGIST both violate their joint dialectical commitment D_3 . In this case, D_1 never detaches, and thus PHYSICIAN is not accountable for the incorrect diagnosis (that is, he does not violate D_1).

What happens if RADIOLOGIST delivers the mammography results on time but her diagnosis is wrong? RADIOLOGIST violates D_2 by delivering an incorrect mammography. PHYSICIAN may incorrectly conclude that PATIENT is free of cancer. In such a case, RADIOLOGIST would be accountable for the erroneous claim.

The questions show how our approach produces models that are valuable for diagnosis and organizational governance.

5 Related Work

Commitments have been extensively employed for modeling processes. However, in contrast to our work, most of the previous work has considered only practical commitments. El Menshawy et al. [7] propose the CTLC+ logic for verifying commitments. Their logic handles practical commitments, and includes modalities for commitment creation and fulfillment (or violation). In contrast, our approach handles both practical and dialectical commitments, and considers the

entire lifecycle of commitments not just their creation and fulfillment (or violation). Specifically, CTLC+ cannot handle scenarios in which the debtor cancels its commitment, or the creditor releases the debtor from the commitment.

Winikoff [27] states that agent interactions designed by focusing on messages restrict agent autonomy by limiting their interaction flexibility. He proposes a commitment-based approach for modeling agent interactions. We agree with Winikoff and employ commitments for modeling processes. However, unlike Winikoff, in addition to practical commitments, we consider dialectical commitments as a first class abstraction to model the guarantees made by the participants (agents). Further, we show how agents' interactions can be verified with respect to their commitments.

Singh [21] presents a combined logic for practical and dialectical commitments. He formulates postulates that capture reasoning patterns for commitments. Our work goes beyond Singh's work in proposing an operationalization of dialectical commitments via a new lifecycle, and showing how to employ CTL to formally verify agent interactions. Additionally, we propose novel reasoning patterns incorporating practical and dialectical commitments.

McBurney and Parsons [13], and Krabbe and Walton [11] describe an argumentation-based representation for agent dialogs (interactions), and formal dialectical systems in argumentation, respectively, that include a notion of commitments. However, their approach violates the autonomy of the participants. For example, a question by one agent may "impose a commitment on the second to provide a response" (p. 266). In contrast, we treat a commitment being created autonomously by its debtor. In addition, we provide a formalization of commitments that supports verification.

Some work on architecture for collaboration is relevant even though it does not incorporate commitments. Narendra et al. [15] propose an architecture framework for modeling cross-enterprise collaborations that consists of three layers: strategy, operational, and service layers. The strategy layer specifies the goals and business rules; the operational layer specifies the services; and the service layer specifies the service implementations. Narendra et al.'s framework lacks adequate modeling of the relationships among the participants. It will be interesting to incorporate commitments (practical and dialectical) to capture the relationships among the participants at the strategy layer.

Liptchinsky et al. [12] propose an approach for modeling dynamic collaboration processes that employs a network of collaborative documents and a social network of collaborators. The notion of relations is a fundamental element in Liptchinsky et al.'s modeling approach. It will be interesting to incorporate commitments to model the relations. Commitments provide a rigorous way to capture the relations among the actors such as an actor (or a group of actors) committing to performing certain action or an actor (or a group of actors) making a claim.

Hofreiter et al. [9] present the UMM methodology for modeling global choreographies, that is, interactions among organizations. UMM seeks to specify a choreography at a high level, independently of the underlying implementation technology. However, UMM lacks well-defined abstractions for capturing the relationships underlying the collaborations. Commitments can provide an abstract

and technology independent way of specifying relationships in UMM’s business domain and requirements views.

We agree with Grando et al. [8] regarding the benefits of high-level abstractions for specifying medical processes. However, unlike our approach, Grando et al. take a centralized viewpoint that violates the autonomy of the participants by mandating their goals. Further, since Grando et al.’s approach ignores the social commitments between the participants, it misses specifying the participants’ responsibilities to each other in the modeled process.

Müller et al. [14] describe the importance of interoperability in healthcare but focus on data interoperability, i.e., with respect to message formats. We incorporate considerations of interactions and thus enable specifying and verifying interoperability in general. For example, a radiologist is interoperable with a hospital not only because they agree on the formats of messages they exchange but because they agree on the commitments involved in those messages.

6 Discussion and Future Work

To model sociotechnical systems, such as service engagements, involves modeling the relevant normative relationships or norms properly [22]. Although we consider commitments as the only norm type in this paper, we give first-class status to dialectical commitments, which are a crucial element of secure collaboration. The main new idea of our approach is highlighting the social nature of dialectical commitments. This idea would readily apply to other norm types. We enhance an existing commitment-based process modeling and verification method [25] to incorporate dialectical commitments and organizational context. In healthcare settings, dialectical commitments enable precisely identifying the accountable party behind a diagnosis.

We incorporate our proposed method into a verification approach and tool based on NuSMV. Our representation enables stating important properties of models in high-level terms to capture stakeholder requirements. Our tool can identify potential errors in models, thereby leading to the design of correct STSs.

In future research, we will address some limitations of this work. In particular, on the theoretical side, we will investigate how dialectical commitments relate to other norm types in STSS from the standpoint of foundations of representing, verifying, and achieving secure collaboration in open settings. On the practical side, we will develop and empirically evaluate an enhanced modeling methodology incorporating dialectical commitments as well as a verification method that incorporates an enhanced notion of time to support better representation and verification of STSs. We will also study how commitments relate to existing business process modeling standards such as BPEL [3].

Acknowledgments. Thanks to the anonymous reviewers for helpful comments and to the US Department of Defense for partial support through a Science of Security Lablet grant.

References

1. ASPE. The importance of radiology and pathology communication in the diagnosis and staging of cancer: mammography as a case study, November 2010. Office of the Assistant Secretary for Planning and Evaluation, U.S. Department of Health and Human Services. <http://aspe.hhs.gov/sp/reports/2010/PathRad/index.shtml>
2. Baldoni, M., Baroglio, C., Chopra, A.K., Singh, M.P.: Composing and verifying commitment-based multiagent protocols. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), pp. 10–17, Buenos Aires, July 2015
3. BPEL. Web services business process execution language, version 2.0, July 2007. <http://docs.oasis-open.org/wsbpel/2.0/>
4. Clarke, E.M., Emerson, E.A., Sistla, A.P.: Automatic verification of finite-state concurrent systems using temporal logic specification. *ACM Trans. Program. Lang. Syst.* **8**(2), 244–263 (1986)
5. Desai, N., Chopra, A.K., Singh, M.P.: Amoeba: a methodology for modeling and evolving cross-organizational business processes. *ACM Trans. Softw. Eng. Methodol. (TOSEM)* **19**(2), 6:1–6:45 (2009)
6. El-Menshawy, M., Bentahar, J., Dssouli, R.: Modeling and verifying business interactions via commitments and dialogue actions. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) *KES-AMSTA 2010, Part II. LNCS*, vol. 6071, pp. 11–21. Springer, Heidelberg (2010)
7. El Menshawy, M., Bentahar, J., El Kholy, W., Dssouli, R.: Reducing model checking commitments for agent communication to model checking ARCTL and GCTL*. *Auton. Agents Multi-Agent Syst.* **27**(3), 375–418 (2013)
8. Grando, M.A., Peleg, M., Glasspool, D.: A goal-oriented framework for specifying clinical guidelines and handling medical errors. *J. Biomed. Inf.* **43**(2), 287–299 (2010)
9. Hofreiter, B., Huemer, C., Liegl, P., Schuster, R., Zapletal, M.: UN/CEFACT's modeling methodology (UMM): a UML profile for B2B e-commerce. In: Proceedings of the 2nd International Workshop on Best Practices of UML (ER), pp. 19–31 (2006)
10. Kalia, A.K., Telang, P.R., Singh, M.P.: Protos: a cross-organizational business modeling tool (demonstration). In: Proceedings of the 11th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), IFAAMAS, pp. 1489–1490, Valencia, Spain, June 2012
11. Krabbe, E.C.W., Walton, D.: Formal dialectical systems and their uses in the study of argumentation. In: Feteris, E.T., Garssen, B., Francisca Snoeck Henkemans, A. (eds.) *Keeping in Touch with Pragma-Dialectics and Computation*, vol.163, pp. 245–263. Benjamin (2011)
12. Liptchinsky, V., Khazankin, R., Schulte, S., Satzger, B., Truong, H.-L., Dustdar, S.: On modeling context-aware social collaboration processes. *Inf. Syst.* **43**, 66–82 (2014)
13. McBurney, P., Parsons, S.: Dialogue games for agent argumentation. In: Simari, G., Rahwan, I. (eds.) *Argumentation in Artificial Intelligence*, pp. 261–280. Springer, USA (2009)
14. Müller, H., Schumacher, M., Godel, D., Khaled Omar, A., Mooser, F., Ding, S.: Medicoordination: a practical approach to interoperability in the Swiss health system. In: *Proceedings of the Medical Informatics in a United and Healthy Europe (MIE)*, vol. 150, pp. 210–214. IOS Press (2009)

15. Narendra, N.C., Lê, L.-S., Ghose, A., Sivakumar, G.: Towards an architectural framework for service-oriented enterprises. In: Ghose, A., Zhu, H., Yu, Q., Delis, A., Sheng, Q.Z., Perrin, O., Wang, J., Wang, Y. (eds.) *ICSOC 2012. LNCS*, vol. 7759, pp. 215–227. Springer, Heidelberg (2013)
16. Norman, T.J., Carbogim, D.V., Krabbe, E.C.W., Walton, D.N.: Argument and multi-agent systems. In: Reed, C., Norman, T.J. (eds.) *Argumentation Machines: New Frontiers in Argument and Computation*, Volume 9 of *Argumentation Library*, Chapter 2, pp. 15–54. Kluwer (2004)
17. NuSMV. A new symbolic model checker (2012). <http://nusmv.fbk.eu>
18. Object Management Group, Framingham, Massachusetts. *UML 2.0 Superstructure Specification*, October 2004
19. Paja, E., Giorgini, P., Paul, S., Meland, P.H.: Security requirements engineering for secure business processes. In: Niedrite, L., Strazdina, R., Wangler, B. (eds.) *BIR Workshops 2011. LNBIP*, vol. 106, pp. 77–89. Springer, Heidelberg (2012)
20. Robinson, W.N., Purao, S.: Specifying and monitoring interactions and commitments in open business processes. *IEEE Softw.* **26**(2), 72–79 (2009)
21. Singh, M.P.: Semantical considerations on dialectical and practical commitments. In: *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI)*, pp. 176–181. AAAI Press, Chicago, July 2008
22. Singh, M.P.: Norms as a basis for governing sociotechnical systems. *ACM Trans. Intel. Syst. Technol. (TIST)* **5**(1), 21:1–21:23 (2013)
23. Singh, M.P.: Cybersecurity as an application domain for multiagent systems. In: *Proceedings of the 14th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), IFAAMAS*, pp. 1207–1212. Blue Sky Ideas Track, Istanbul, May 2015
24. Telang, P.R., Kalia, A.K., Singh, M.P.: Engineering service engagements via commitments. *IEEE Internet Comput.* **18**, 1–8 (2014)
25. Telang, P.R., Singh, M.P.: Specifying and verifying cross-organizational business models: an agent-oriented approach. *IEEE Trans. Serv. Comput.* **5**(3), 305–318 (2012). Appendix pp. 1–5
26. Verdicchio, M., Colombetti, M.: Commitments for agent-based supply chain management. *SIGecom Exchan.* **3**(1), 13–23 (2002)
27. Winikoff, M.: Designing commitment-based agent interactions. In: *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp. 363–370 (2006)