Commitment-based business processes

Fabiano Dalpiaz, John Mylopoulos,
Paolo Giorgini, Amit K. Chopra

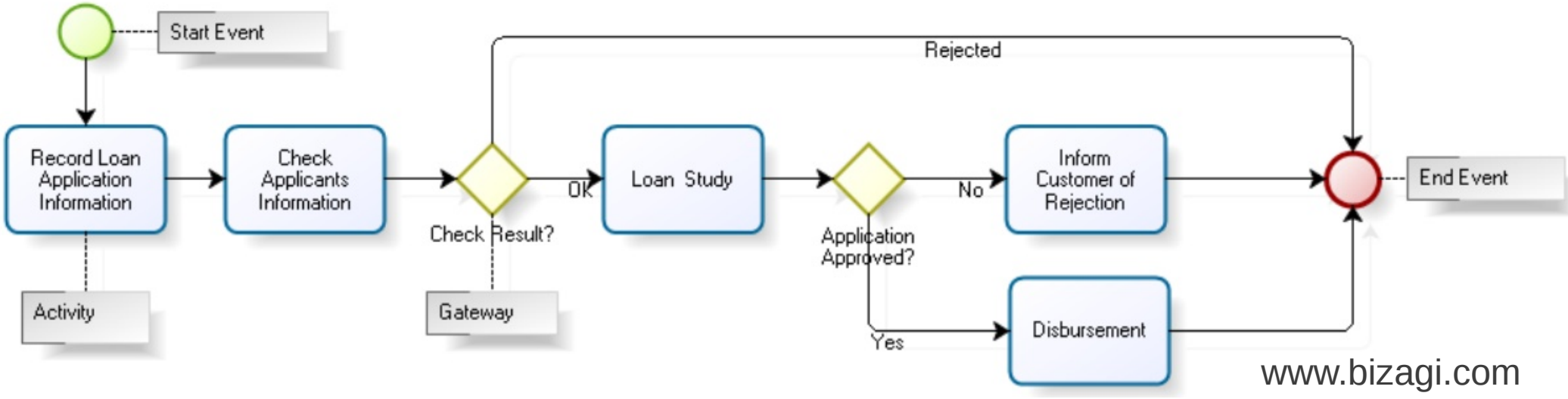
January 20, 2012

Business processes modelling and execution

- **Business processes (BPs)** define how business has to be conducted within an organization
  - Remark: not just within an organization: inter-organizational BPs
- **BP modelling** is concerned with the representation of BPs
  - Typically, as a set of interrelated activities
- Some BP modelling notations natively support **BP execution**
  - Which activities are to be performed and by whom
  - **Work-flow engines** enable BP execution

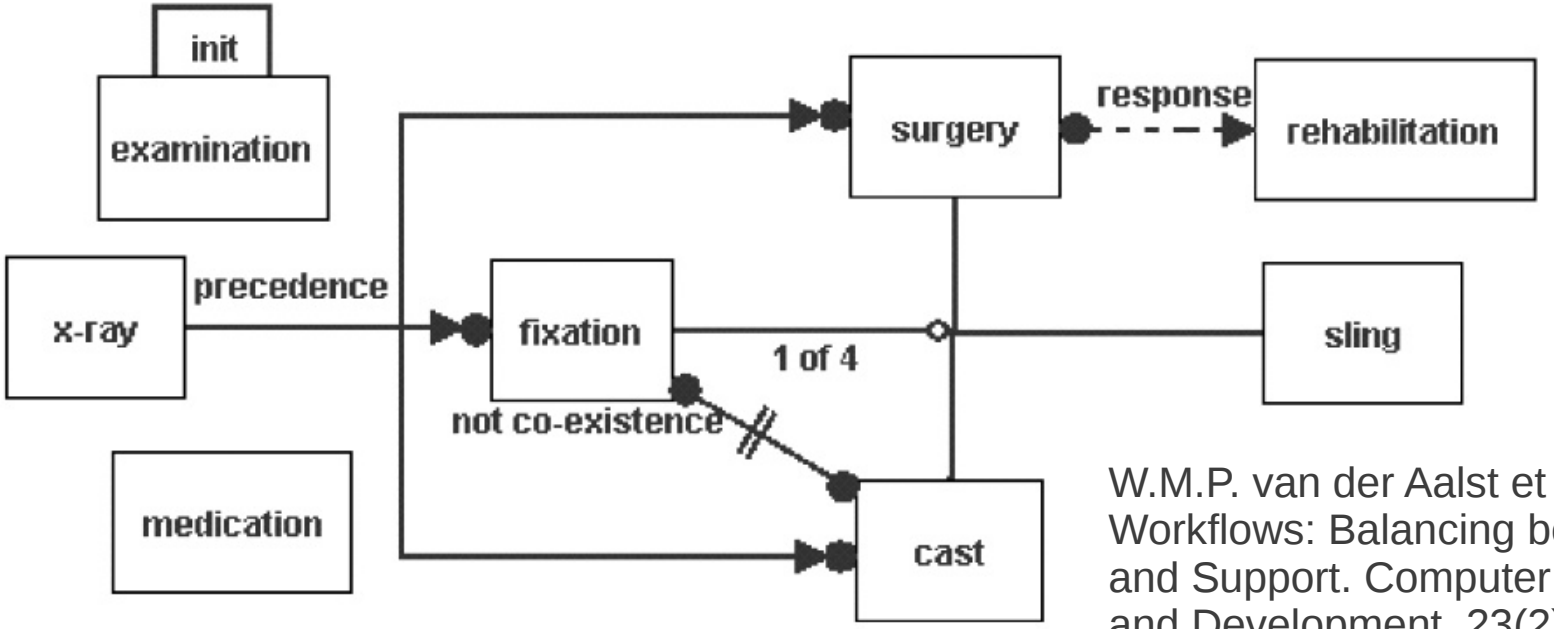
Work-flow centric BP modelling

- A BP is modelled in terms of **activities** and **control flow**



- BP execution: a work-flow engine assigns activities to performers
  - Performers have no freedom on the activities to execute
  - Exception handling: change performers, follow a different path

Declarative BP modelling

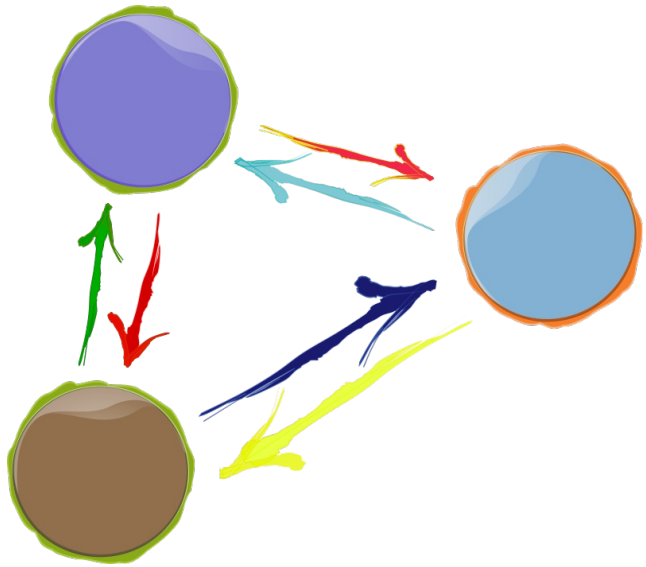


W.M.P. van der Aalst et al. Declarative Workflows: Balancing between Flexibility and Support. Computer Science-Research and Development, 23(2):99–113, 2009.

+ **more flexibility** (precedence constraints, no control flow)
+ possibility to create custom links based on LTL
- defined in terms of **activities**

BPs are interaction-centric

- A BP is a **multi-party interaction among participants**
  - The sequencing of activities is a consequence of the data flow
  - Participants are interested in **fulfilling their commitments**...
  - … but want to maintain **freedom** on how they conduct themselves



BPs are interaction-centric

- Example: take the commitments of doctor Sarah in a hospital
  - C_1 (to the head clinician): “make the daily round”
  - C_2 (to nurse Mara): “visit patient Tom within noon”
  - C_3 (to the secretary): “fill in weekly report within 4PM”
- As long as she fulfils C_1, C_2, and C_3, she can act freely!

Modelling language: key concepts

- **Agent**: fabiano, paolo, tong, elda, vitor, univTrento, ...
- **Role**: Professor, MS Student, PhD Student, Post-doc
- **Commitment**: promise with contractual validity from a debtor to a creditor that, if an antecedent is brought about, a consequent will be brought about
  - Antecedent and consequent are **states of affairs**
  - c(vitor, fabiano, seminarDetailsSent, seminarAnnounced)
  - In our language, **we consider commitments between roles**
  - **C(SeminarOrganizer, Presenter, DetailsSent, Announced)**

Syntax (by example)

Protocol parameters: agents that are bound when the protocol is Instantiated. Among them, the context agent

protocol Treatment (**context** h, p : Patient, sp : Specialist) {
ag-variables: rc, ra, or;
commitments:
init $\rightarrow_t$ C_1:C(sp, p, $\top$, Examined $\cdot$ Diagnosed)
NoXRayNeeded $\rightarrow_t$ C_2:C(Orthopedist, sp, $\top$, SlingMade)
XRayRequested $\rightarrow_t$ C_3:C(Radiologist, sp, $\top$, bind(deb, ra) $\cdot$ XRayPerformed)

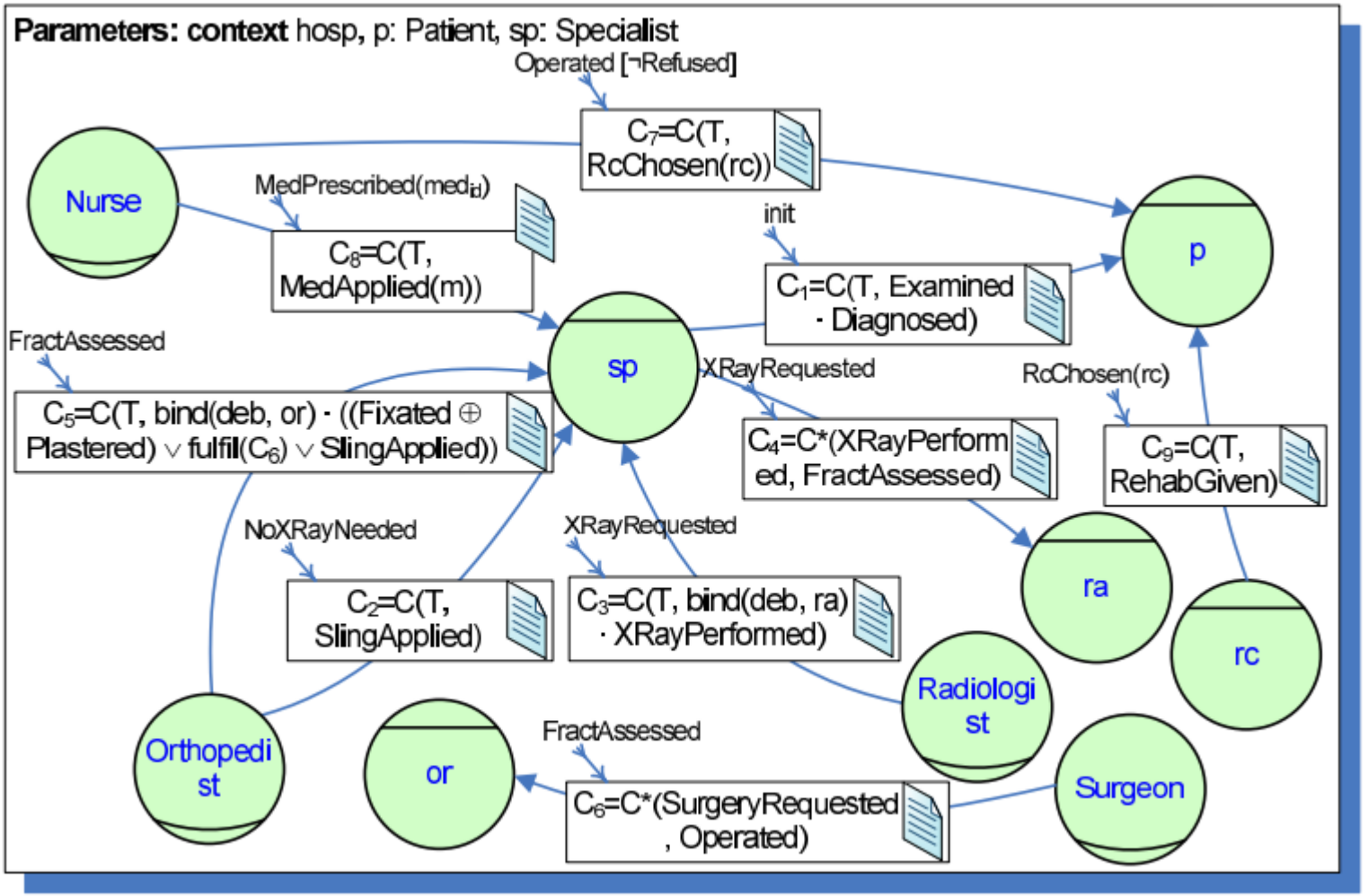
Precedence operator "$\cdot$"

As soon as the protocol is instantiated

As the LHS happens, the commitment in the RHS shall be created

The debtor is bound to agent variable "ra"

Preliminary graphical syntax



Advanced syntax: commitment refinements

- **Deadline**: the commitment shall be fulfilled within a time-out
- **Authorizations**
  - **Delegation** of a commitment to a different agent (by the debtor)
    - Responsibility: retained vs. released
  - **Assignment** of a commitment to another agent (by the creditor)
    - Credit: retained vs. released
  - **Cancellation** of a commitment (by the debtor)

Advanced syntax: constraints & more

- **Cardinality constraints per role**
  - max number of concurrent commitment instances
  - max number of concurrent commitment instances of a given commitment class
    - e.g. the doctor cannot commit to visit more than 3 patients
- **Separation of duties**
  - Two commitments shall have different debtors
- **Compensation**
  - If a commitment is violated, another shall be brought about
  - e.g. if the doctor damages a patient's leg, he'll have to refund him

Modelling language: semantics

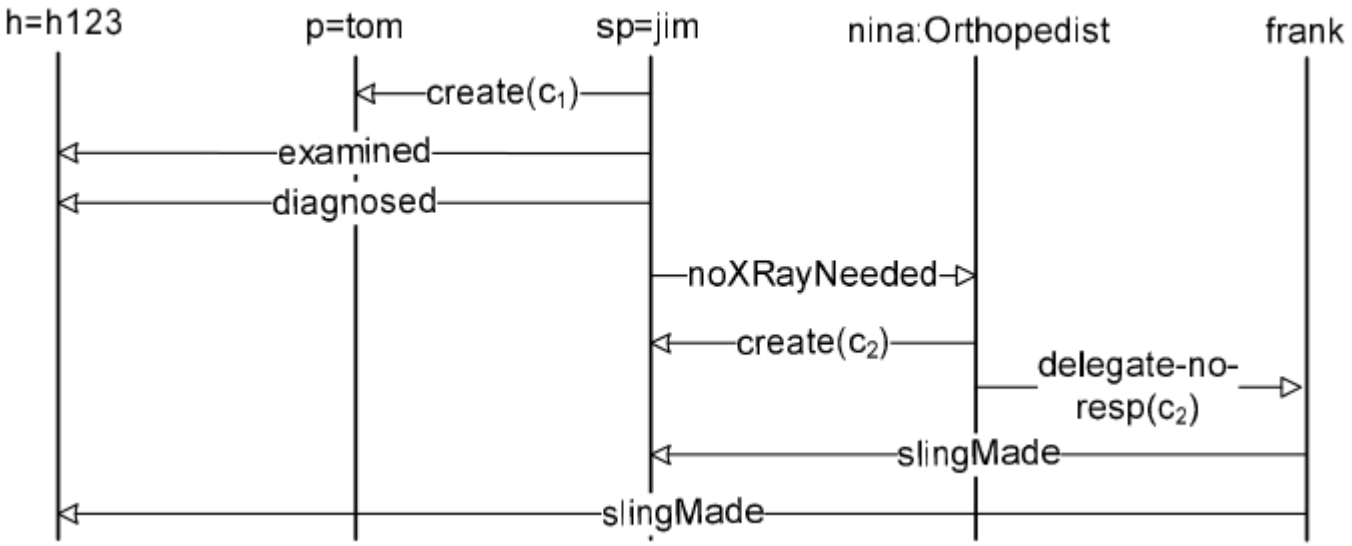
- Key points:
  - **All language primitives are commitments!**
  - The semantics defines constraints on the execution
  - Agents are free to act, as long as they fulfil their commitments
- Intuition 1 (deadline)

  The debtor commits to the creditor that, as soon as the the commitment is detached (it becomes unconditional), the consequent will be brought about within the set deadline
- Intuition 2 (compensation)

  The debtor commits to the creditor that, if the commitment is violated, the compensation commitment will be created

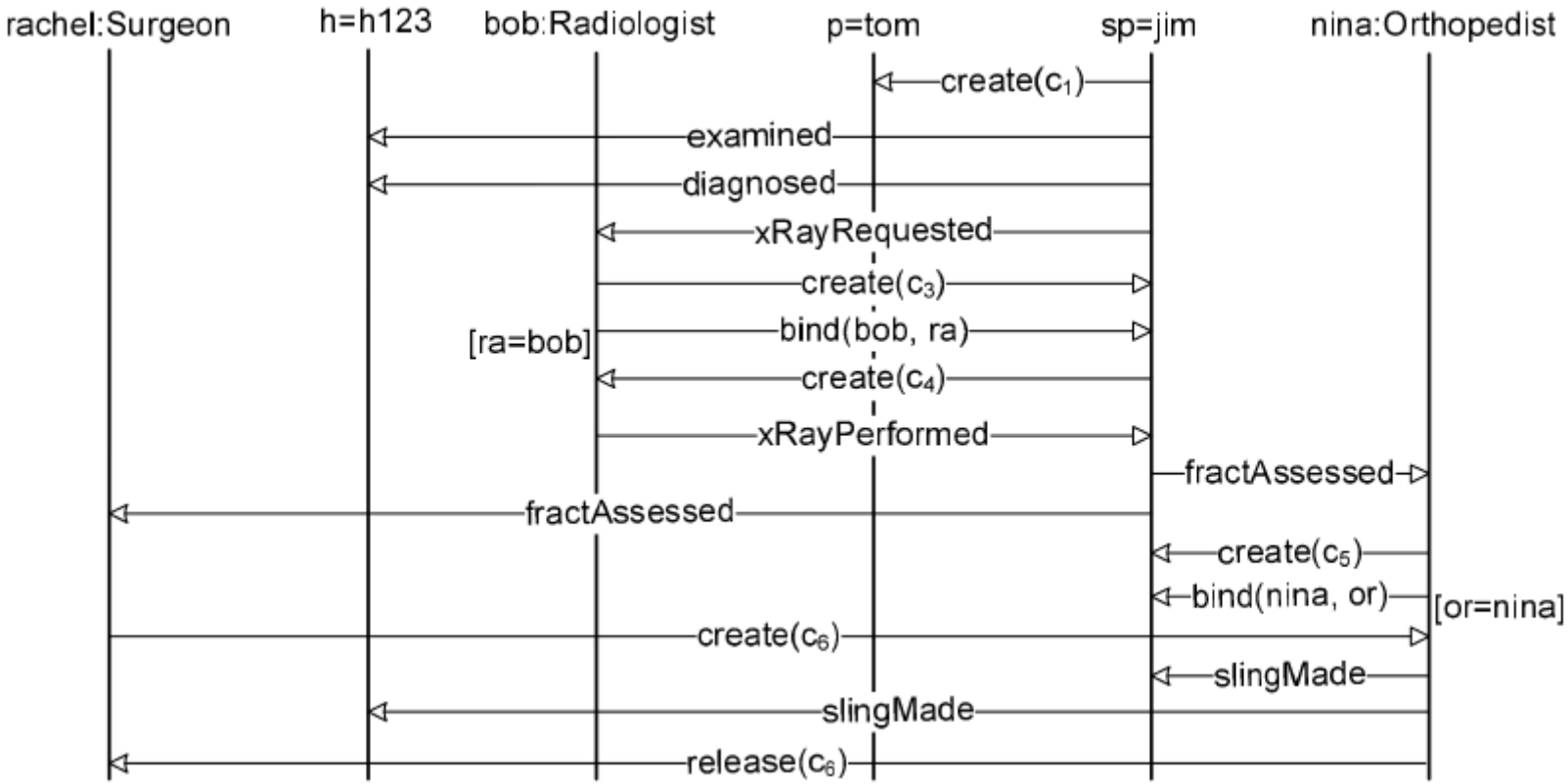
Executing (enacting) a protocol

- A protocol enactment is a set of exchanged messages!



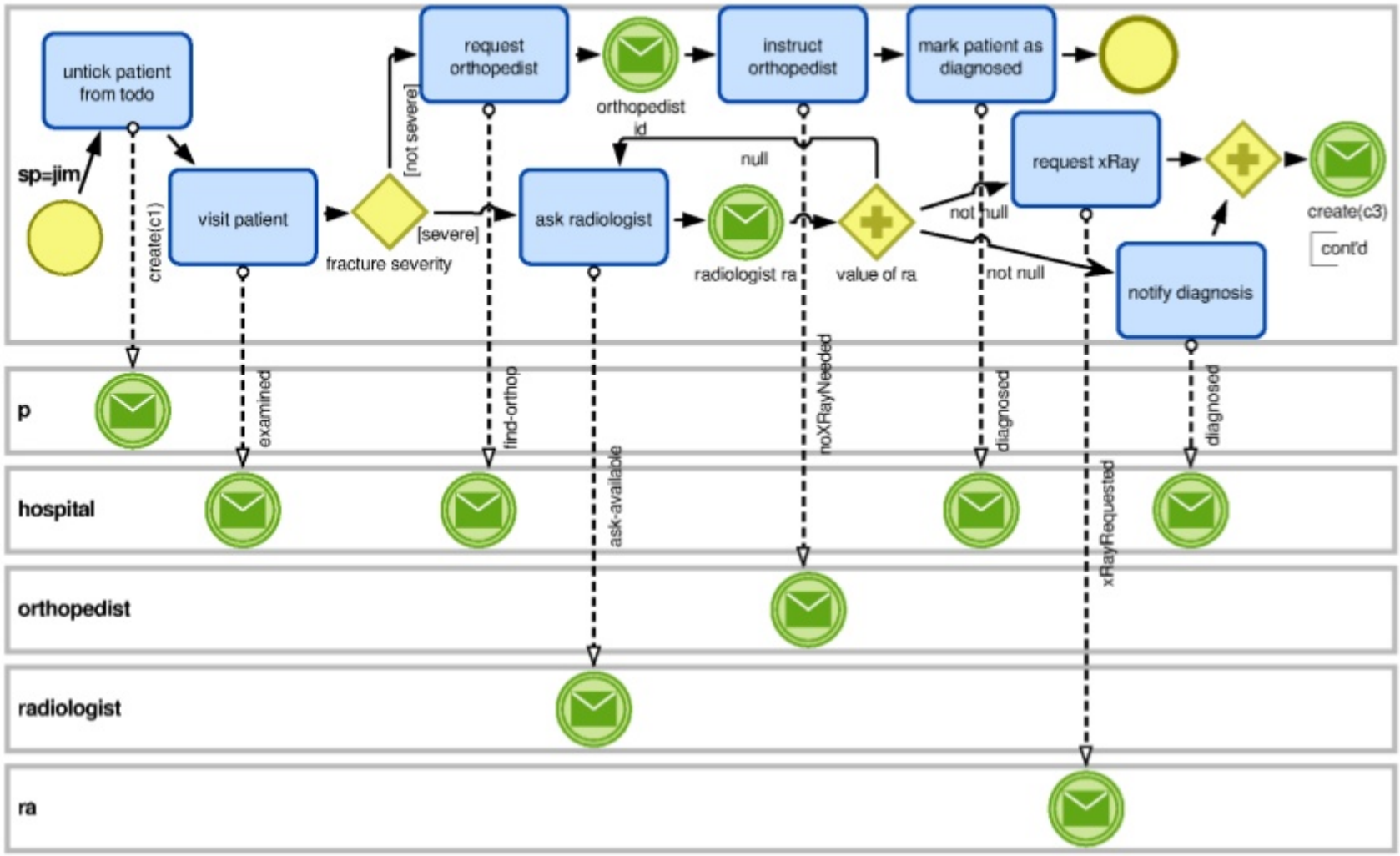
Enactment 1: Xrays are not needed, a sling is made!

An alternative enactment



Enactment 2: Xrays are needed; however, just a sling is made!

Work-flows have a place: an agent's perspective



Jim is a specialist; his work-flow can be defined, e.g., as activities and messages (BPMN 2.0)

Conclusions

- Activity-based BP modelling and execution is too inflexible
- BPs are **situated social activities** (see CSCW)
  - Thus, interaction among parties is first-class
- Our language is centred around the notion of **commitments**
  - The agents are free to act, as long as they comply with their commitments
  - **Decoupling** between an **agent**'s construction and execution and the **process** (protocol) specification and enactment
- **Remark**: ongoing work, not written in stone