Comparing business models using graph-edit distance

Ivanov Sergey,
research assistant

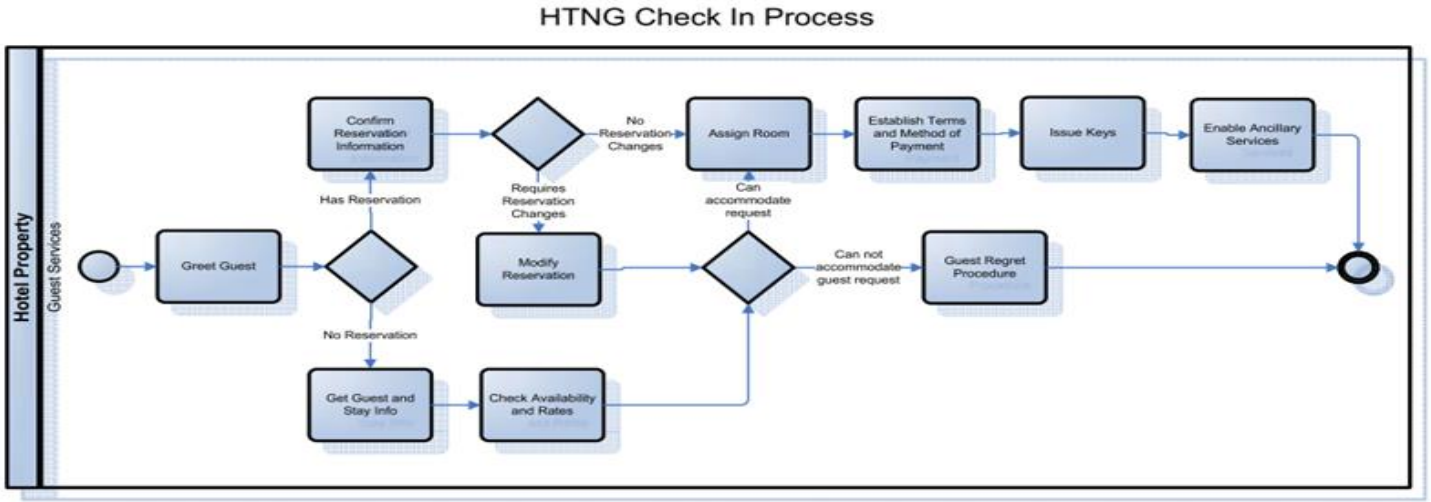
Agenda

- Business process models (BP)
- Changes in BPs
- BPMN
- BPMN model as a graph
- Graph-edit distance
- Our approach for comparing two BP models

- 40 mins
- Q&A on the spot

BP models

Business process modeling (BPM) in systems engineering is the activity of representing processes of an enterprise, so that the current process may be analyzed or improved.

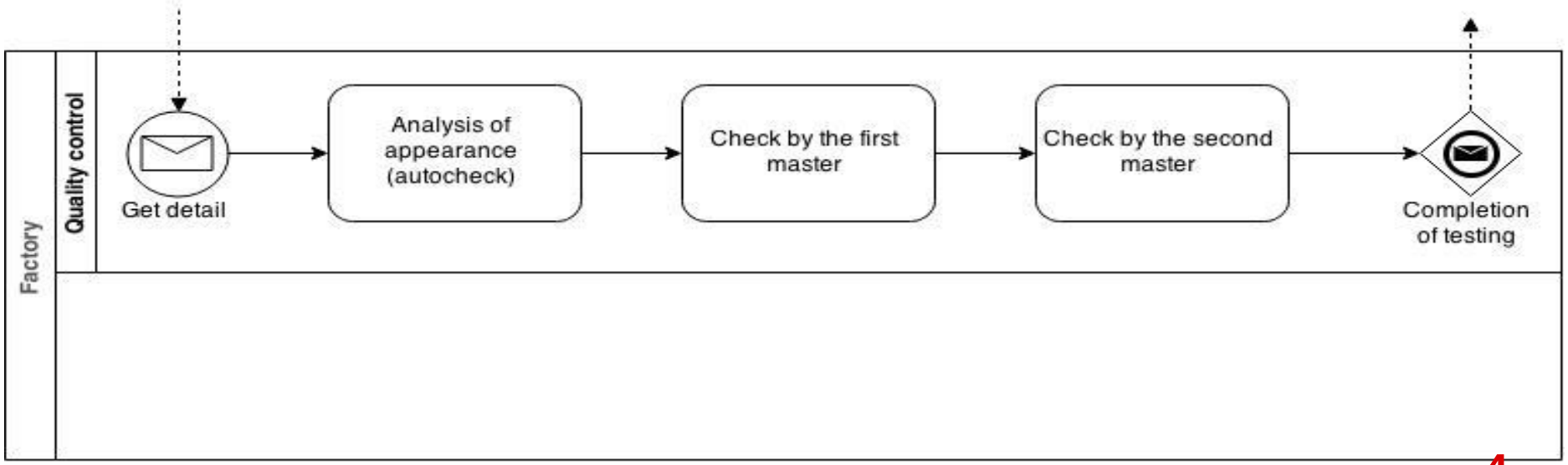


Changes in business processes

- new amendments to the legislation
- changes in revenue sources
- increase production efficiency
- economic progress
- technological progress

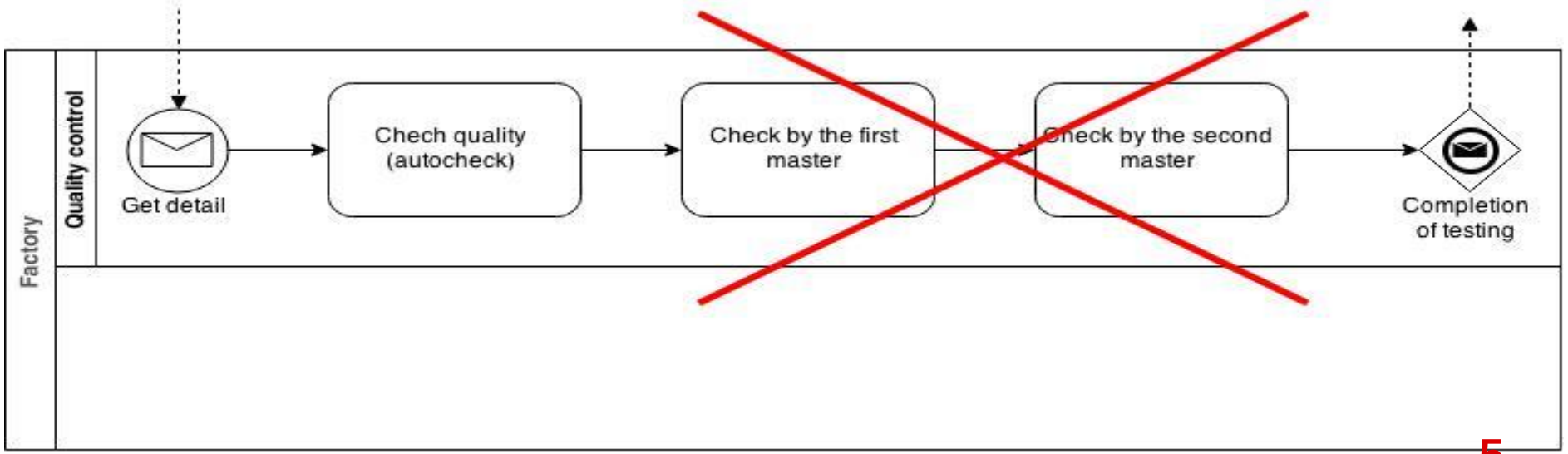
Example 1-1

Double quality control by two masters



Example 1-2

Purchase of equipment for testing quality of details

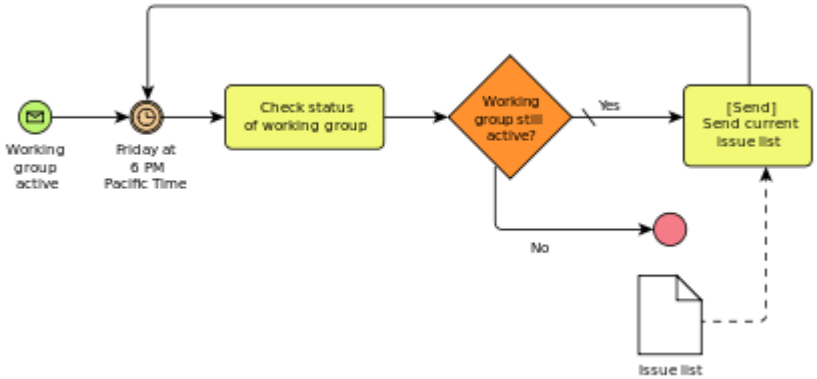


Business process modeling tools

- BPMN - Business Process Model and Notation
- CogNIAM - Cognition enhanced Natural language Information Analysis Method
- xBML - Extended Business Modeling Language
- EPC - Event-driven process chain
- IDEF0 - ICAM DEFinition
- UML - Unified Modeling Language

BPMN

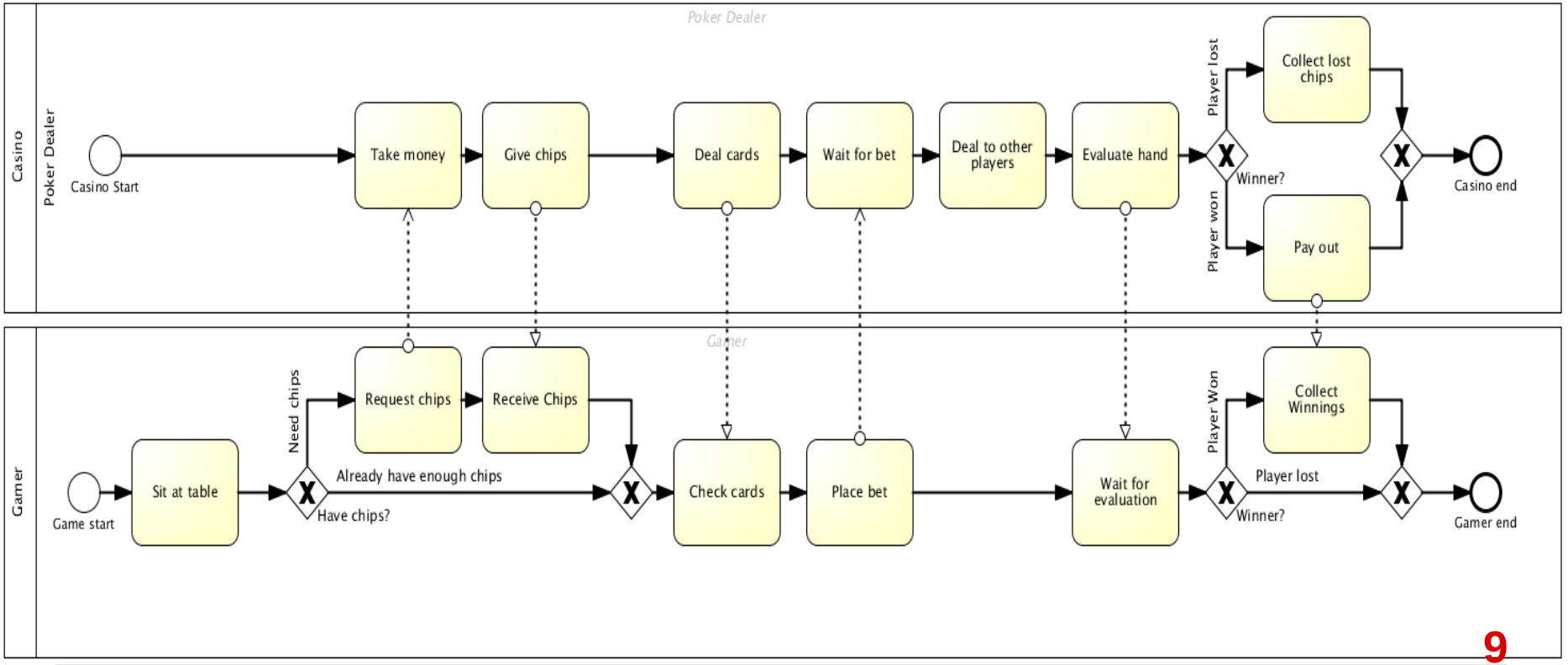
Business Process Model Notation (BPMN) is a graphical representation for specifying business processes in a business process model.



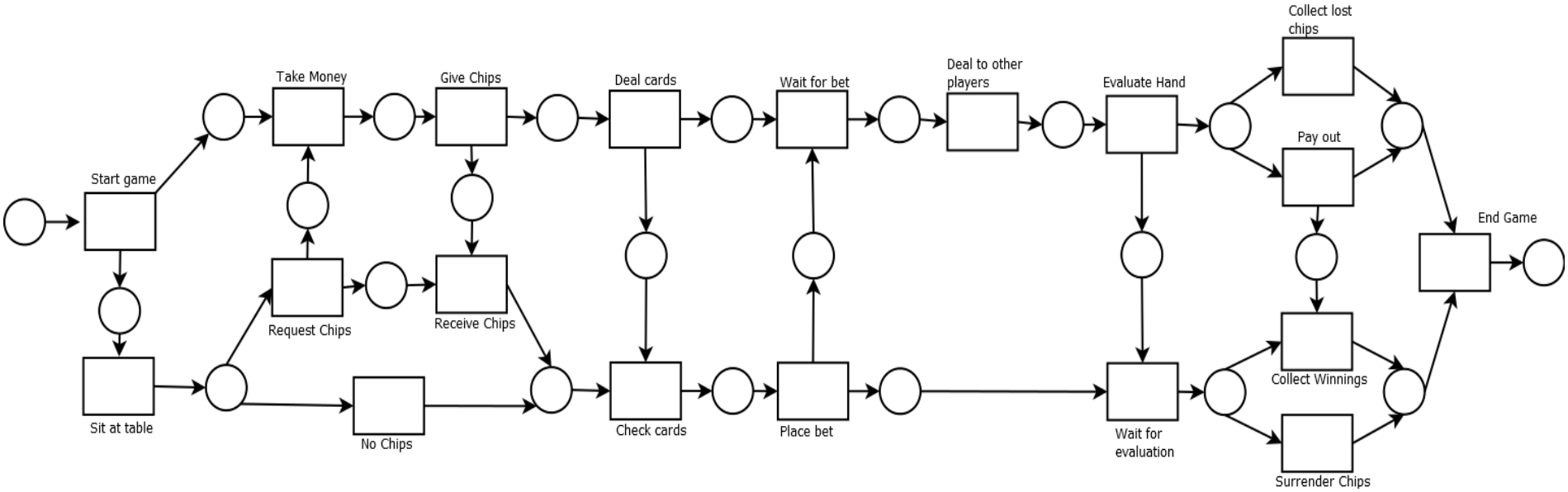
BPMN features

- Must be acceptable and usable by the business community for general process modeling
- Must be able to generate executable processes from a BPMN Model (through a combination of graphical elements and supporting information)
- BPMN is intended to be Methodology

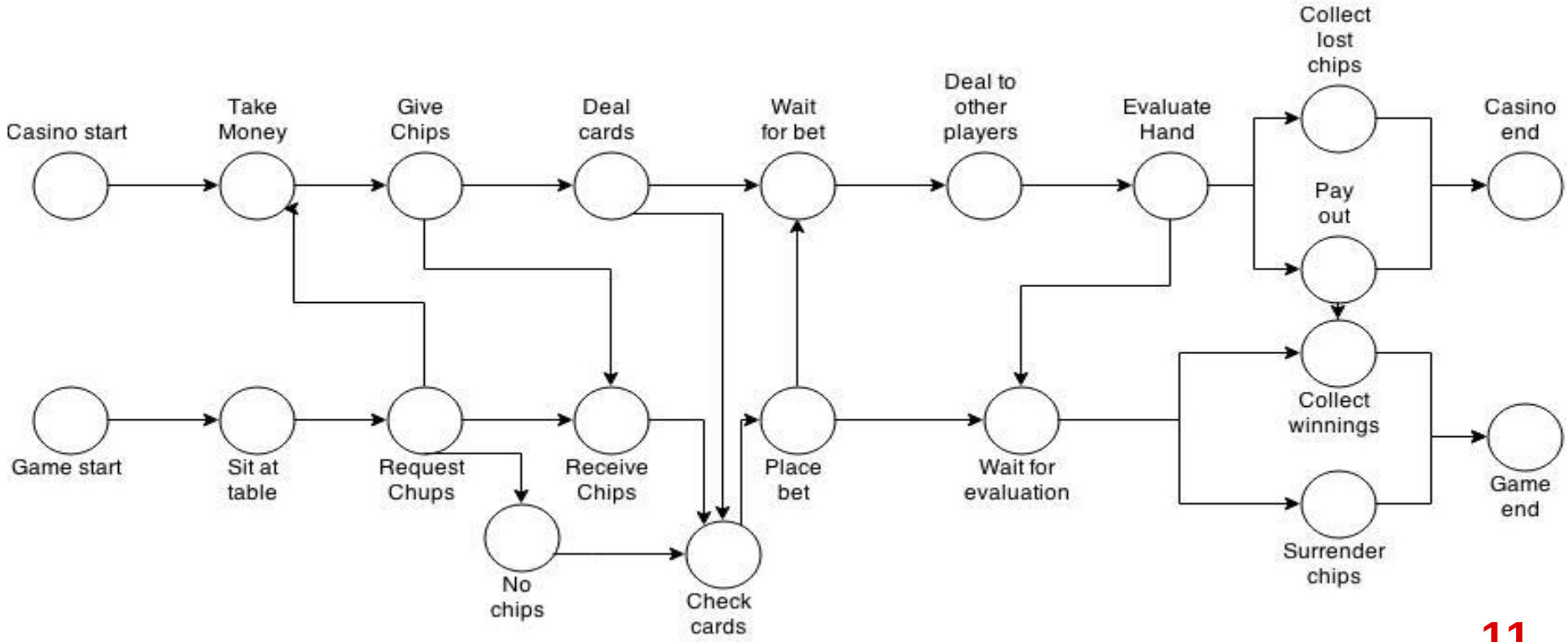
BPMN model


Casino
Poker Dealer
Poker Dealer
Casino Start
Take money
Give chips
Deal cards
Wait for bet
Deal to other players
Evaluate hand
Winner?
Player lost
Collect lost chips
Player won
Pay out
Casino end
Gamer
Gamer
Game start
Sit at table
Need chips
Request chips
Receive Chips
Already have enough chips
Have chips?
Check cards
Place bet
Wait for evaluation
Player Won
Collect Winnings
Player lost
Winner?
Gamer end

BPMN model as a net

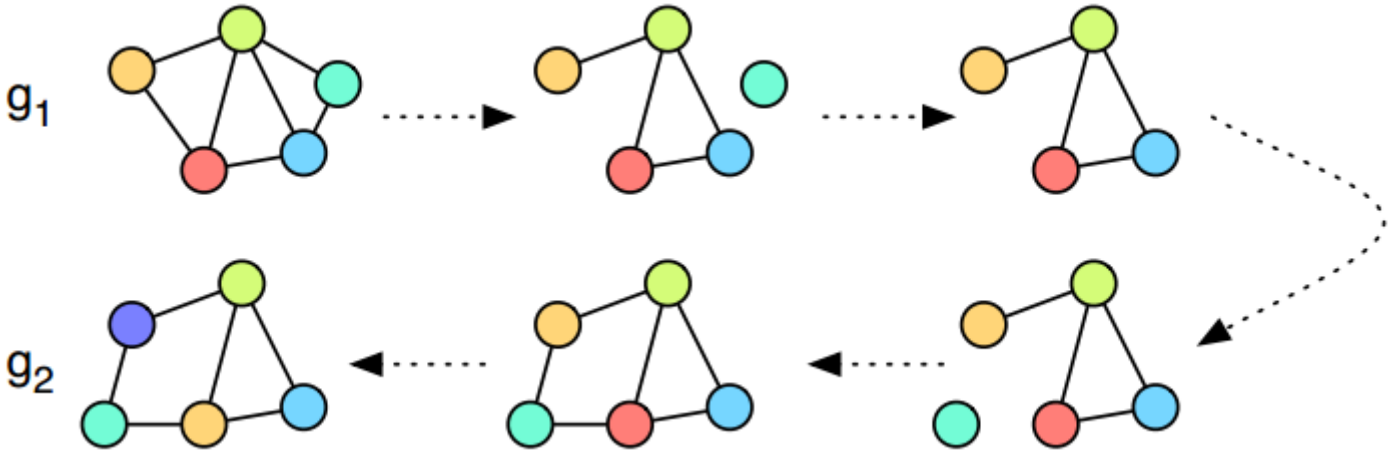


BPMN model as a graph



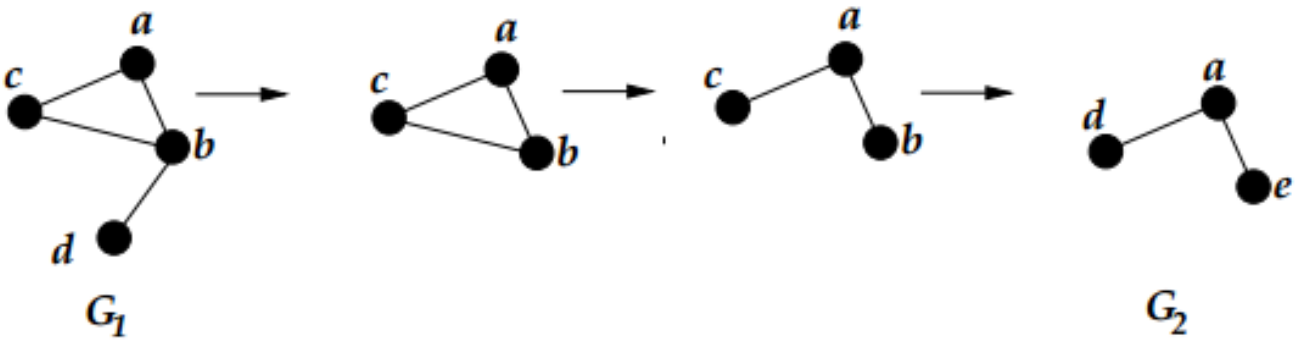
Comparing graphs

Calculate graph edit distance using A* algorithm



Graph edit distance

The graph edit distance *d* of two graphs *g1* and *g2* is the minimum costs equence of *edit operations* (next slide) that transform *g1* in to *g2*.



Graph edit operations

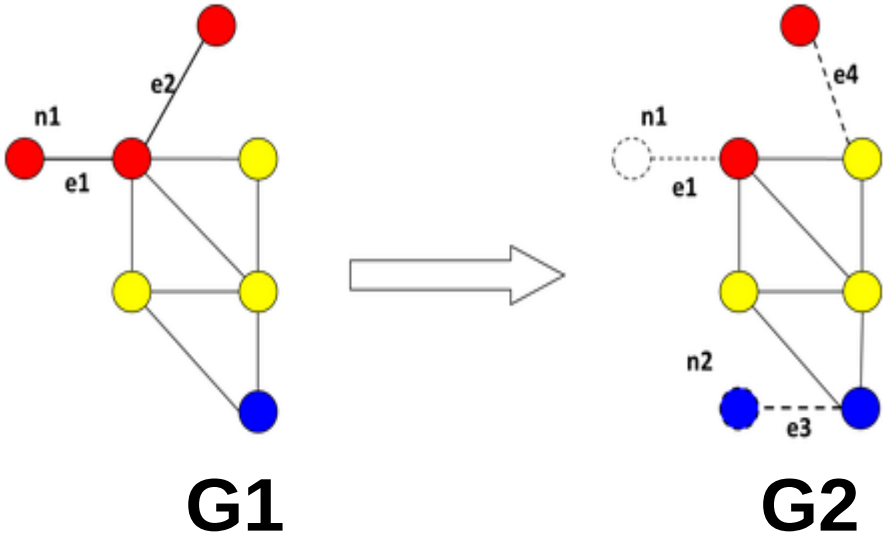
A graph edit operation is an edit operation to transform one graph to another, including 6 types:

- Insert an isolated vertex into the graph
- Delete an isolated vertex from the graph
- Change the label of a vertex
- Insert an edge between two disconnected vertices
- Delete an edge from the graph
- Change the label of an edge

Example 2-1

Cost of edit operations:

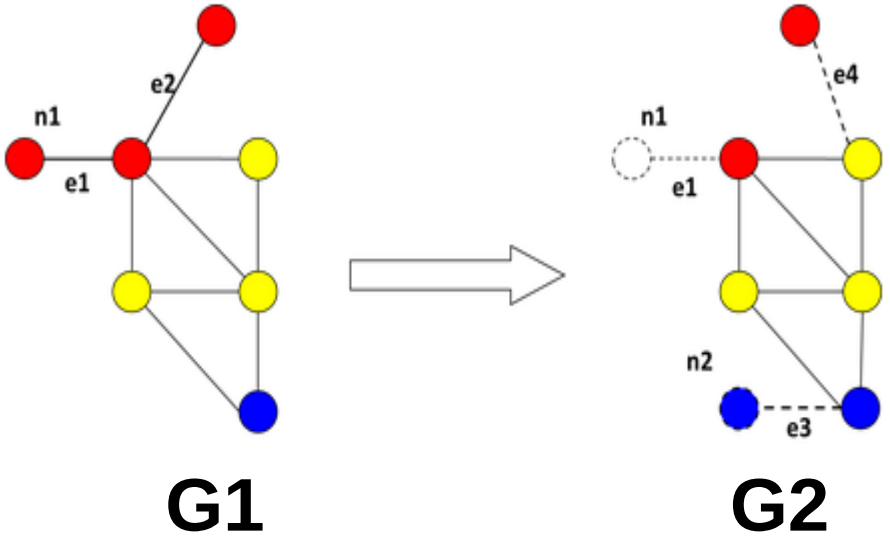
- deletion/insertion of edges/nodes – 1
- substitution of nodes/edges – 0.



Example 2-2

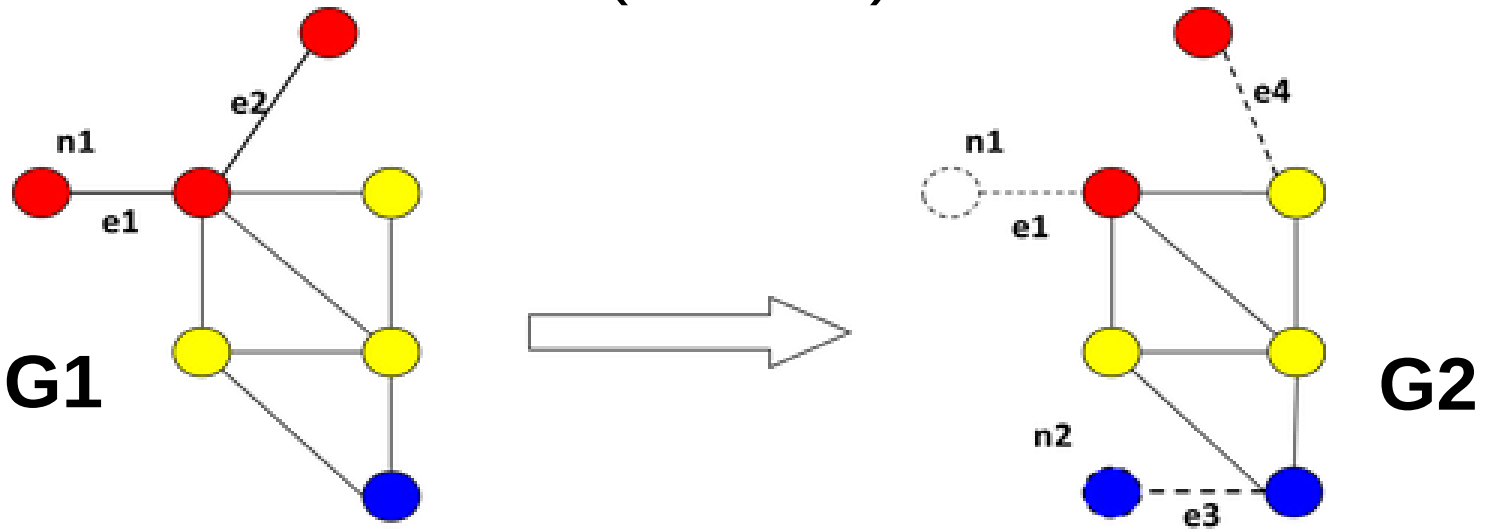
Transforming:

- delete edge **e1**
- delete node **n1**
- delete edge **e2**
- insert edge **e4**
- insert node **n2**
- insert edge **e3**



Example 2-3

GED(**G1**, **G2**) = 6



Graph edit distance

Two main drawbacks:

- a high computational complexity: the problem of computing graph edit distance is NP-hard in general
- the difficulty related to defining cost functions

Possible solutions:

- the most known method for computing the exact value of graph edit distance is based on A$*$ search algorithm
- experimental evaluations

Graph edit distance by A* algorithm

Algorithm 1. Computation of graph edit distance by A* algorithm

Input: Non-empty graphs $g_1 = (V_1, E_1, \mu_1, \nu_1)$ and $g_2 = (V_2, E_2, \mu_2, \nu_2)$, where $V_1 = \{u_1, \ldots, u_{|V_1|}\}$ and $V_2 = \{v_1, \ldots, v_{|V_2|}\}$

Output: A minimum-cost edit path from g_1 to g_2 e.g. $p_{min} = \{u_1 \to v_3, u_2 \to \varepsilon, \ldots, \varepsilon \to v_6\}$

1: Initialize OPEN to the empty set
2: For each vertex $w \in V_2$, insert the substitution $\{u_1 \to w\}$ into OPEN
3: Insert the deletion $\{u_1 \to \varepsilon\}$ into OPEN
4: **loop**
5: Remove $p_{min} = \arg\min_{p \in \text{OPEN}} \{g(p) + h(p)\}$ from OPEN
6: **if** p_{min} is a complete edit path **then**
7: Return p_{min} as the solution
8: **else**
9: Let $p_{min} = \{u_1 \to v_{i_1}, \ldots, u_k \to v_{i_k}\}$
10: **if** $k < |V_1|$ **then**
11: For each $w \in V_2 \setminus \{v_{i_1}, \ldots, v_{i_k}\}$, insert $p_{min} \cup \{u_{k+1} \to w\}$ into OPEN
12: Insert $p_{min} \cup \{u_{k+1} \to \varepsilon\}$ into OPEN
13: **else**
14: Insert $p_{min} \cup \bigcup_{w \in V_2 \setminus \{v_{i_1}, \ldots, v_{i_k}\}} \{\varepsilon \to w\}$ into OPEN
15: **end if**
16: **end if**
17: **end loop**

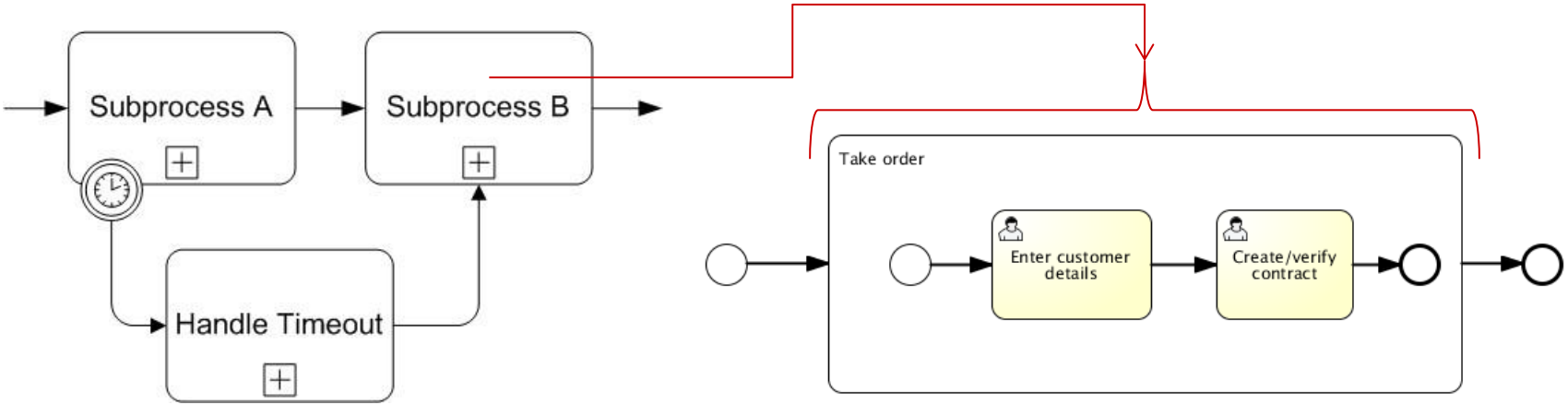
Comparing BPMN models

Differences from the comparison graphs:

- each element has a label
- each element has a special type (event, activity, gateway, connection)
- subprocess like subgraphs

Comparing BPMN models

A subprocess element should be represented as a subgraph which means that nested elements should be considered as separate elements.



Comparing BPMN models

Comparing two elements:

- string-edit distance for labels
- comparing types

Comparing two models:

- string-edit distance + comparing types
- A* algorithm

Comparing two BPMN models:

- string-edit distance + *preprocessing*
- A* algorithm + high-level comparison

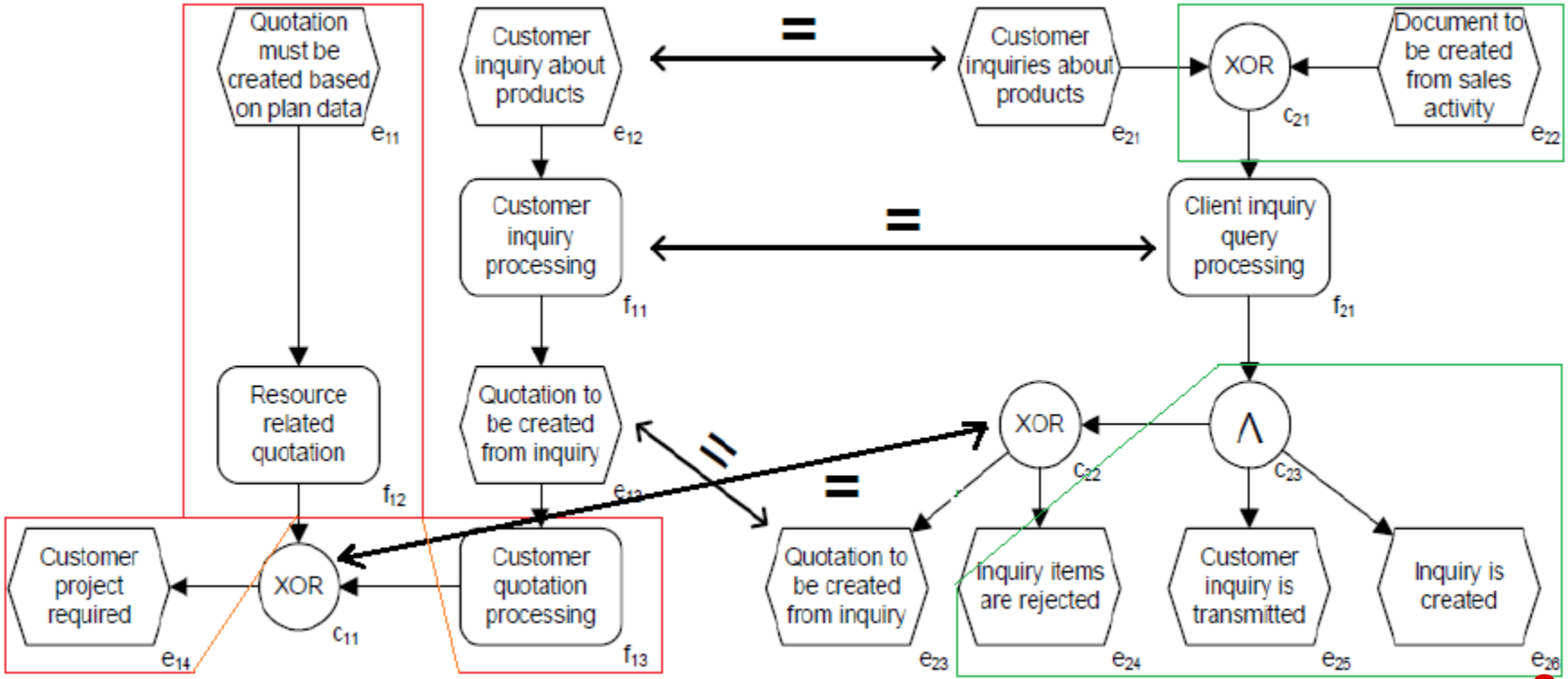
String edit distance distance

The Levenshtein distance is a string metric for measuring the difference between two sequences.

Operations: insertion, deletion or substitution.

1. kitten → sitten (substitution of "s" for "k")
2. sitten → sittin (substitution of "i" for "e")
3. sittin → sitting (insertion of "g" at the end).

Example 3-1

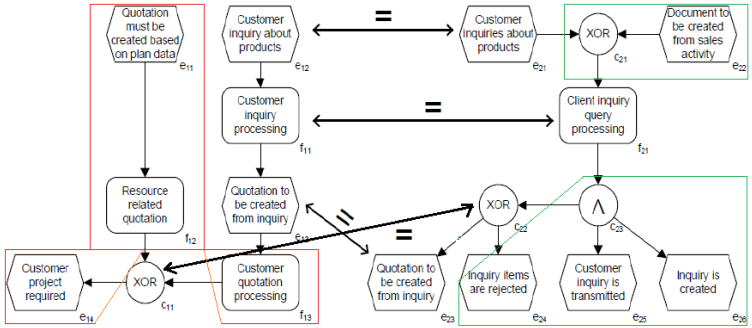

Quotation must be created based on plan data
e11
Customer inquiry about products
e12
Customer inquiries about products
e21
XOR
c21
Document to be created from sales activity
e22
Customer inquiry processing
f11
Client inquiry query processing
f21
Resource related quotation
f12
Quotation to be created from inquiry
e13
XOR
c22
∧
c23
Customer project required
e14
XOR
c11
Customer quotation processing
f13
Quotation to be created from inquiry
e23
Inquiry items are rejected
e24
Customer inquiry is transmitted
e25
Inquiry is created
e26

Example 3-2

Graph edit distance = 14

Actions:

- delete event 1
- insert activity 5
- change label for event 3
- …



Thanks a lot
for your attention!