Build a better life by translating problems to Datalog and Answer Set Programming(ASP)

Axel Polleres

Institute for Data, Process and Knowledge Management

28/11/2022 Invited Talk - TAASP 2022, Vienna, Austria

Social media: @AxelPolleres@wien.rocks

http://polleres.net/

NOVEMBER 2022

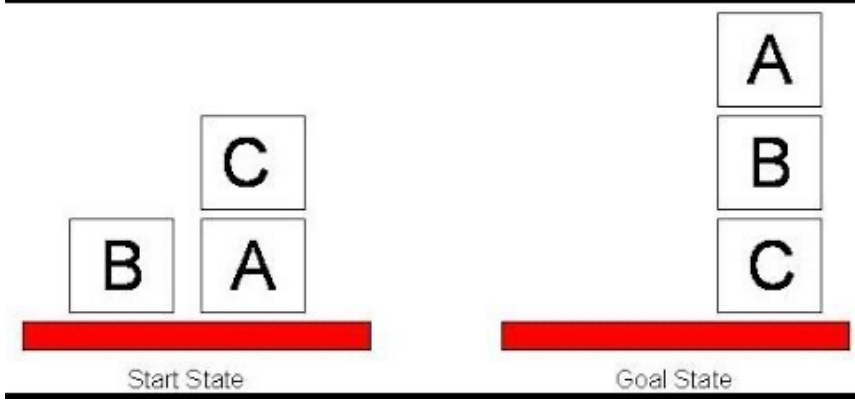
Why I love ASP (since over 20 years)…

- ***Intuitive***, understandable Problem ***encodings***…`
- … easily ***extensible***
- the beauty of **Guess and Check** to solve complex problems on top

Station 1:
1999-2003

AI Planning

AI Planning…



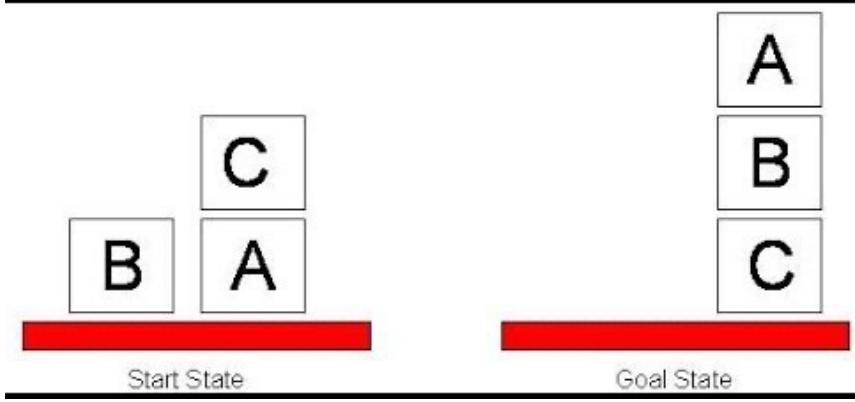
- fluents: **`on(c,a), on(b,table), on(a,table)`**
- Actions: **`move(c,table)`** with
  - preconditions:

```
executable move(B, L) if block(B).
nonexecutable move(B, B).
nonexecutable move(B, L) if blocked(B).
nonexecutable move(B, L) if blocked(L).
```

  - effects:

```
caused on(B, L) after move(B, L).
caused -on(B, L) after move(B, L1), on(B,L), L1<>L.
```

AI Planning… intuitive encoding



- fluents: `on(c,a,0). on(b,table,0). on(a,table,0).`
- Actions: `move(c,table,T).`
  - preconditions:

```
:- move(B,B,T).
:- move(B,L,T) , blocked(B,T).
:-move(B,L) , blocked(L,T) .
```

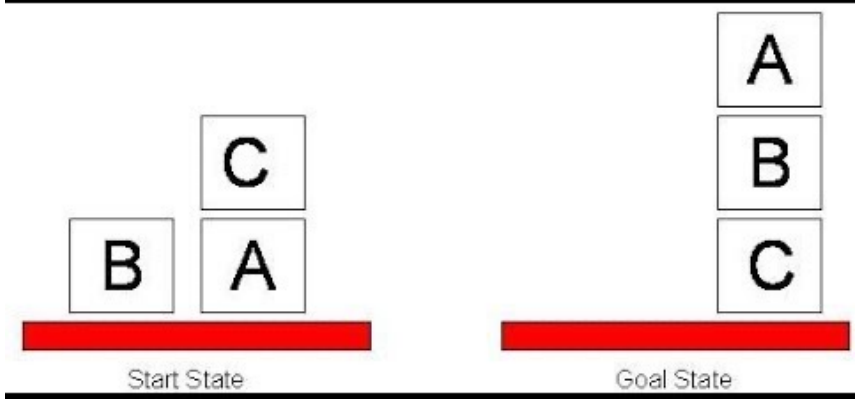
  - effects:

```
on(B, L, T+1) :- move(B, L, T).
-on(B, L, T+1) :- move(B, L1, T), on(B,L,T), L1<>L.
```

  - implicit background knowledge:

```
blocked(B,T) :- on(_,B,T), block (B).
on(B,L,T+1) :- on(B,L,T), not -on(B,L,T).
```

AI Planning… intuitive encoding



- fluents: `on(c,a,0). on(b,table,0). on(a,table,0).`
- Actions: `move(c,table,T).`
  - preconditions:

```
:- move(B,B,T).
:- move(B,L,T) , blocked(B,T).
:-move(B,L) , blocked(L,T) .
```

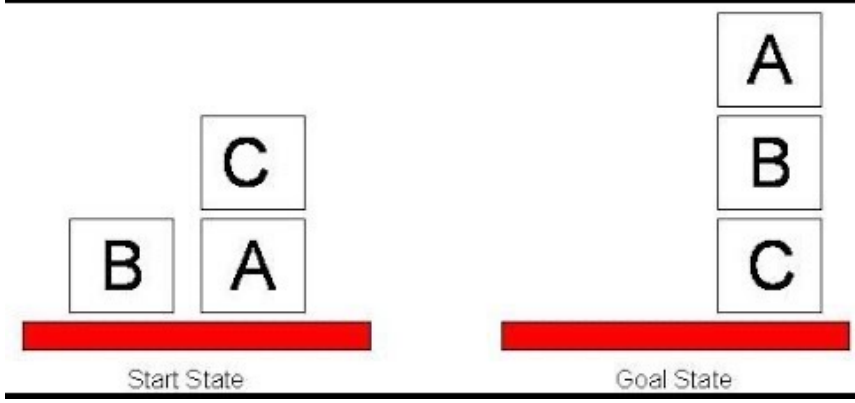
  - effects:

```
on(B, L, T+1) :- move(B, L, T).
-on(B, L, T+1) :- move(B, L1, T), on(B,L,T), L1<>L.
```

  - implicit background knowledge:

```
blocked(B,T) :- on(_,B,T), block (B).
on(B,L,T+1) :- on(B,L,T), not -on(B,L,T).
```

AI Planning… guess and check:



- fluents: `on(c,a,0). on(b,table,0). on(a,table,0).`
- Actions: `move(c,table,T).`
  - preconditions:

    move(B, L,T) v -move(B, L,T) :- block(B), location(L), time(T).

    ```
    :- move (B,B,T).
    :- move(B, L,T) , blocked(B,T).
    :- move(B, L) , blocked(L,T) .
    ```

    ```
    blocked(B,T) :- on(_,B,T).
    ```

  - effects:

    ```
    on(B, L, T+1):- move(B, L,T).
    -on(B, L,T+1):- move(B, L1,T), on(B,L,T), L1<>L.
    ```

  - implicit background knowledge:

    ```
    blocked(B,T) :- on(B1,B,T), block (B).
    on(B,L,T+1) :-on(B,L,T), not -on(B,L,T).
    ```

Goal:

```
goal(T) :- on(a,b),
           on(b,c),
           on(c,table).
:- not goal(maxTime).
time(0..maxTime).
```

AI Planning… ease of extensions:

- Nondeterministic actions:
  - E.g., “clumsy” move

```
on(B, L, T+1) v on(B, table, T+1) :- cmove(B, L, T), on(B,L,T), block(L).
on(B, L, T+1) :- cmove(B,L,T), on(B,table,T).
```

- Action costs /cost optimal planning:

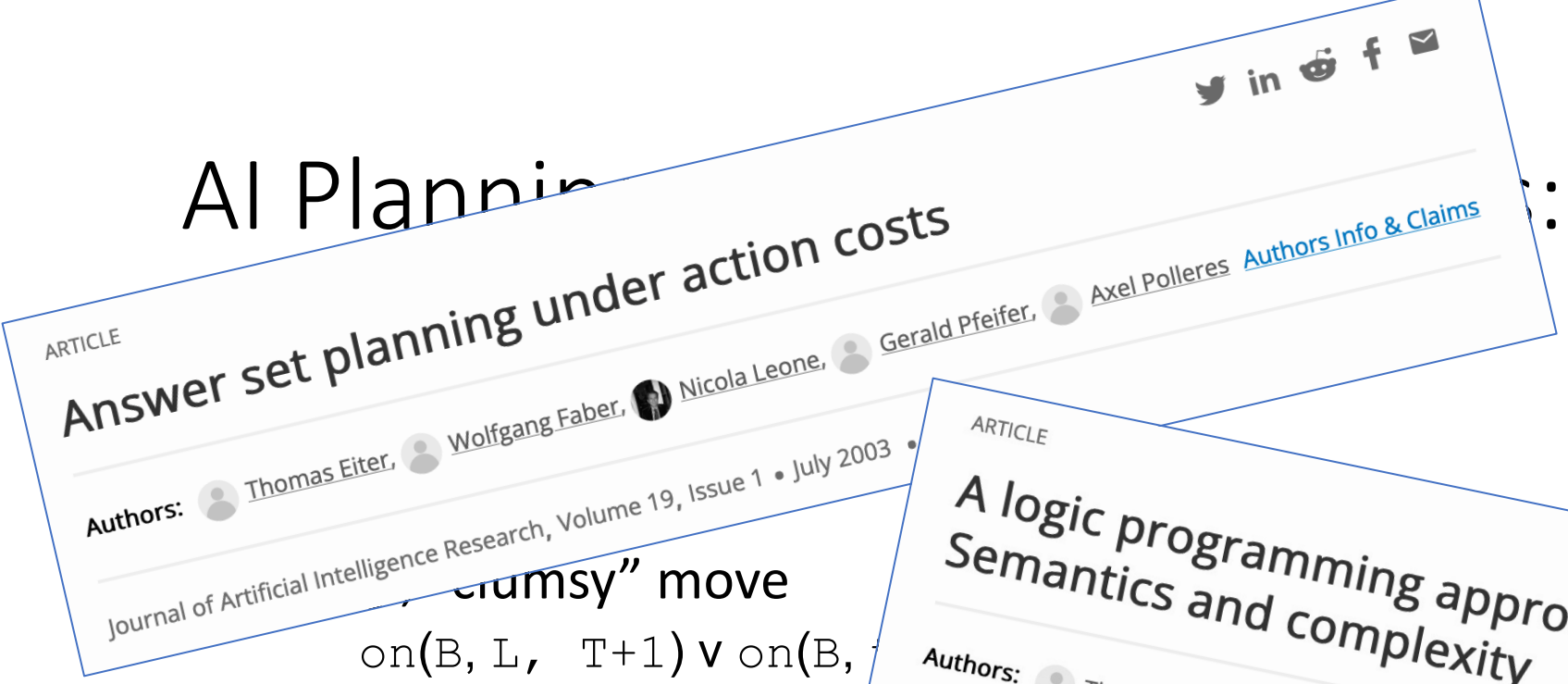
```
cost(T,1) :- cmove(B,L,T).
cost(T,2) :- move(B,L,T).
:~ cost(T,Cost). [Cost:]
```

AI Plannin

clumsy" move

```
on(B, L, T+1) v on(B,
on(B, L, T+1) :-
```

- Action costs /cost optimal



Why I love ASP (since over 20 years)…

- ***Intuitive***, understandable Problem ***encodings***…`
- … easily ***extensible***
- the beauty of **Guess and Check** to solve complex problems on top

Part 1:
1999-2003

AI Planning

Part 2:
2003 – to date…

Universidad
Rey Juan Carlos

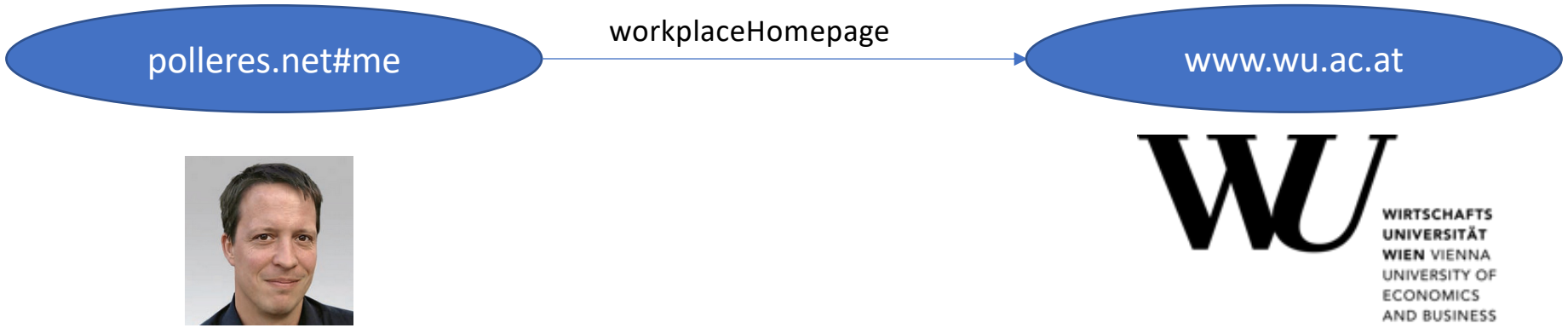
Semantic Web

Semantic Web … RDF Triples

- “Typed” links on the Web …

```
<http://www.polleres.net#me> <http://xmlns.com/foaf/0.1/workplaceHomepage> <http://www.wu.ac.at> .
```

- … can be seen as subject-predicate-object edges in a Graph:



Semantic Web … Standards like RDF have lead to (really) big Open “Knowledge Graphs”…

- … availabe on the Web
- … ***queryable via a query language called SPARQL***!

1,101,215,718 triples/edges

13,602,048,837 triples/edges

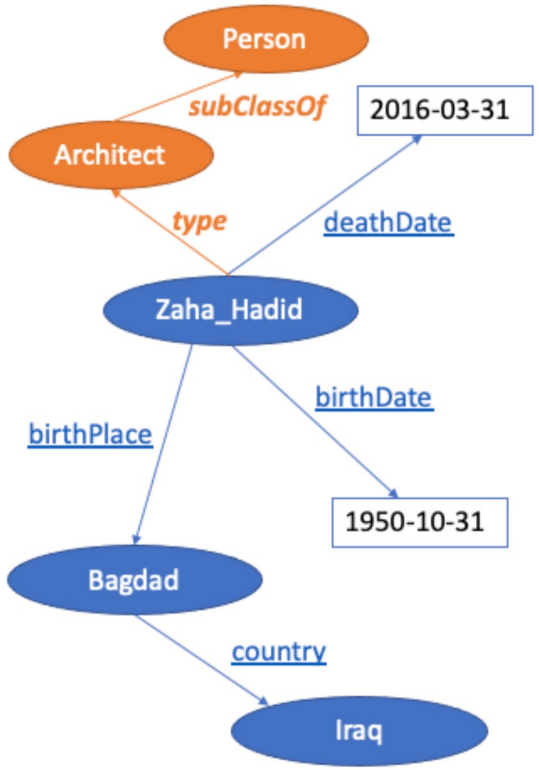
Semantic Web … Standards like RDF have lead to (really) big Open “Knowledge Graphs”…

- … availabe on the Web
- … ***queryable via a query language called SPARQL***!

http://wikipedia.org/wiki/Zaha_Hadid

Born	Zaha Mohammad Hadid 31 October 1950 Baghdad, Kingdom of Iraq
Died	31 March 2016 (aged 65) Miami, Florida, U.S.
Nationality	Iraq, United Kingdom
Alma mater	American University of Beirut Architectural Association School of Architecture
Occupation	Architect
Parent(s)	Mohammed Hadid Wajeeha Sabonji
Practice	Zaha Hadid Architects
Buildings	Vitra Fire Station, MAXXI, Bridge Pavilion, Contemporary Arts Center, Heydar Aliyev Center, Riverside Museum
Website	www.zaha-hadid.com

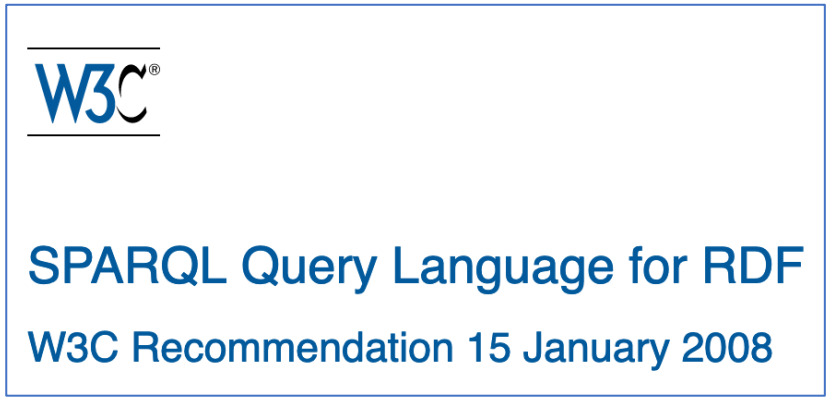
http://dbpedia.org/resource/Zaha_Hadid



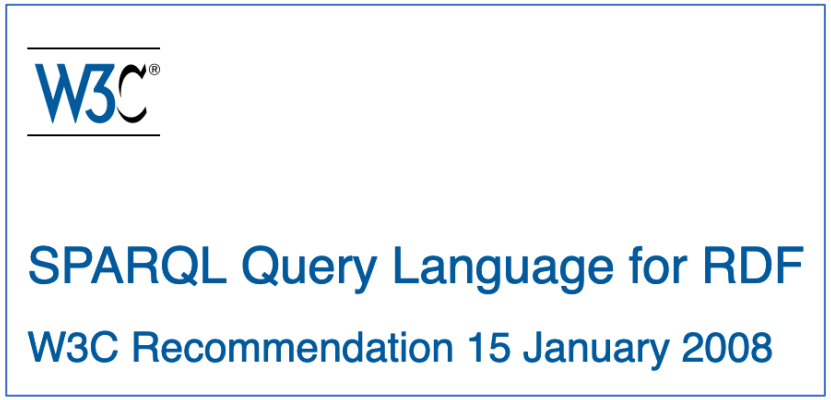

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?N WHERE {
  ?X rdf:type dbo:Architect ;
     rdfs:label ?N .
  {{?X dbo:birthPlace dbr:Baghdad}
   UNION
   {?X dbo:birthPlace dbr:Vienna}}

  FILTER (lang(?N) = "en")

}
```

Semantic Web …Status ~2006

- Semantics of SPARQL in parts undefined
- Various extensions being discussed…



SPARQL… intuitive encoding



- Why I liked SPARQL?

```
SELECT ?X
WHERE {
   ?X rdf:type dbpedia:Architect.
   ?X dbpedia:birthPlace  dbpedia:Baghdad .
      }
```

for the same reason I love ASP! *Obvious similarities to Datalog…*

```
answer(X) :-
        triple( X,  birthPlace , baghdad ) ,
        triple( X,  type , architect) .
```

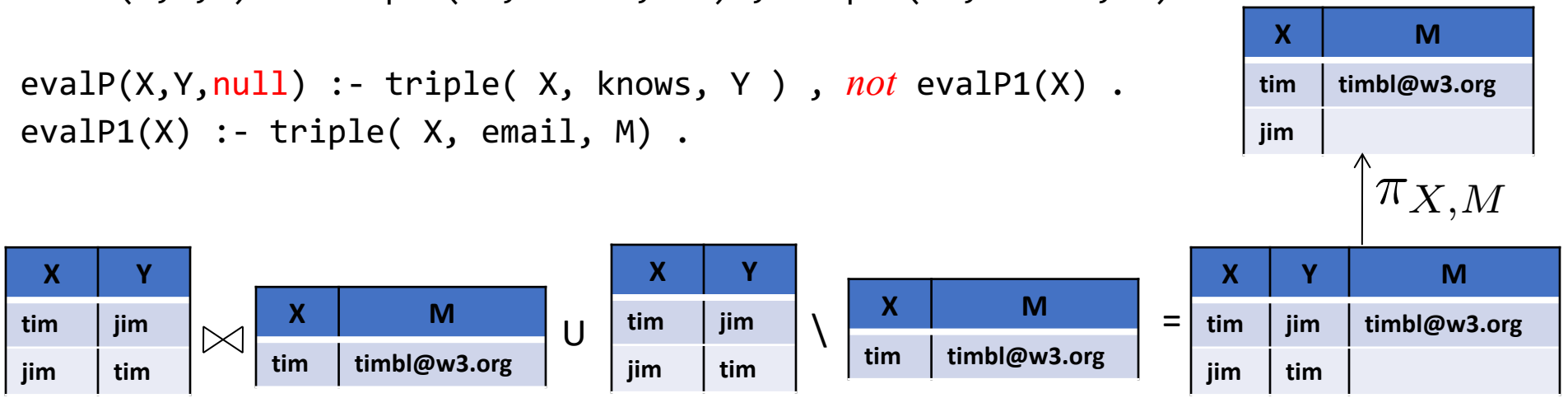
SPARQL… intuitive encoding some not entirely trivial, e.g. OPTIONAL:

Give me people who know somebody and OPTIONALLY their email address:

```
triple( :tim, knows, :jim ) .   triple(:tim, email, timbl@w3.org ) .
triple( :jim, knows, :tim ) .
```

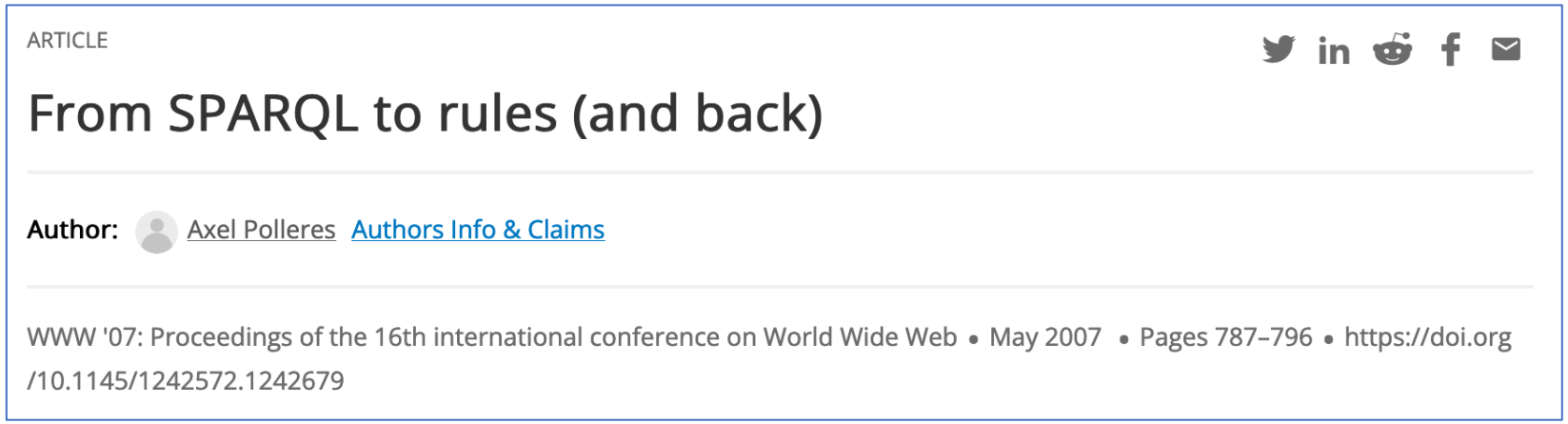
Example Query:

```
answer(X,M)  :- evalP(X,Y,M) .

evalP(X,Y,M) :- triple( X, knows, Y ) , triple( X, email, M) .

evalP(X,Y,null) :- triple( X, knows, Y ) , not evalP1(X) .
evalP1(X) :- triple( X, email, M) .
```



SPARQL… intuitive encoding

[Polleres, 2007] shows that all of SPARQL 1.0 can be translated to (safe) nonrecursive Datalog^{not}.



ARTICLE

From SPARQL to rules (and back)

Author: Axel Polleres Authors Info & Claims

WWW '07: Proceedings of the 16th international conference on World Wide Web • May 2007 • Pages 787–796 • https://doi.org/10.1145/1242572.1242679

[Angles&Gutierrez 2008] vice versa show that (safe) nonrecursive Datalog^{not} likewise be encoded into SPARQL.

ARTICLE

The Expressive Power of SPARQL

Authors: Renzo Angles, Claudio Gutierrez Authors Info & Claims

ISWC '08: Proceedings of the 7th International Conference on The Semantic Web • October 2008 • Pages 114–129 • https://doi.org/10.1007/978-3-540-88564-1_8

SPARQL… ASP's ease of extensions:

1) We could show that additional features of SPARQL 1.1 were also easily encodable in Datalog…


Articles

On the relation between SPARQL1.1 and Answer Set Programming

Axel Polleres & Johannes Peter Wallner

Pages 159-212 | Published online: 24 Jun 2013

Download citation https://doi.org/10.1080/11663081.2013.798992

2) … and (already before) proposed a semantics to using SPARQL as a rules language to define the semantics of RDF sources potentially mutually referring to each other…


ARTICLE

SPARQL++ for mapping between RDF vocabularies

Authors: Axel Polleres, François Scharffe, Roman Schindlauer Authors Info & Claims

OTM'07: Proceedings of the 2007 OTM Confederated international conference on On the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS - Volume Part I • November 2007 • Pages 878–896

SPARQL… ASP's ease of extensions:

ARTICLE

SPARQL++ for mapping between RDF vocabularies

Authors: Axel Polleres, François Scharffe, Roman Schindlauer Authors Info & Claims

OTM'07: Proceedings of the 2007 OTM Confederated international conference on On the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS - Volume Part I • November 2007 • Pages 878–896

Unfortunately didn't make it to the standard ;-), but here's the idea:

Web source P1:

```
:tim :knows :jim .
:tim :email <mailto:timbl@w3.org> .
:jim :knows :tim .

CONSTRUCT {?s :email ?m }
FROM P2
WHERE {?s :email ?m }
```

Web source P2:

```
:jim <mailto:jim@rpi.edu> .

CONSTRUCT {jim :knows ?m }
FROM P1
WHERE {:jim :knows ?m }
```

SPARQL… ASP's guess and check :

ARTICLE

SPARQL++ for mapping between RDF vocabularies

Authors: Axel Polleres, François Scharffe, Roman Schindlauer Authors Info & Claims

OTM'07: Proceedings of the 2007 OTM Confederated international conference on On the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS - Volume Part I • November 2007 • Pages 878–896

Unfortunately didn't make it to the standard ;-), but here's the idea:

Web source P1:

```
:tim :knows :jim .
:tim :email <mailto:timbl@w3.org> .
:jim :knows :tim .

CONSTRUCT {?s :email ?m }
FROM P2
WHERE {?s … OPTIONAL … }
```

Web source P2:

```
:jim <mailto:jim@rpi.edu> .

CONSTRUCT {jim :knows ?m }
FROM P1
WHERE {:jim … NOT EXISTS … }
```

Could involve recursion and cycles over negation, elegantly solvable with ASP semantics!

Semantic Web … more recent ASP applications:

- SHACL: a language to formulate constraints over RDF graphs

Shapes Constraint Language (SHACL)

W3C Recommendation 20 July 2017

- E.g.:

Shape:

```
:StudentShape a sh:NodeShape ;
    sh:targetNode :Ben ;
    sh : property [
        sh:path :enrolledIn ;
        sh : qualifiedMinCount 1 ;
        sh : qualifiedValueShape [
            sh:class :Course ]] .
```

Each defined target node needs to fulfill the constraints, in this case:

There needs to be at least one Course that the student is enrolled in.

Data Graph:

```
: Ben :enrolledIn :C1 .
:C1 :type Course.
```

SHACL … computing repairs by Guess and check:

- Idea: Without going into details of the encoding: we encode repairs inspired by

Database Repairing and Consistent Query Answering
Synthesis Lectures on Data Management
August 2011, 121 pages, (https://doi.org/10.2200/S00379ED1V01Y201108DTM020)

Leopoldo Bertossi
Carleton University, Ottawa, Canada

SHACL … computing repairs by Guess and check:

- Idea: Without going into details of the encoding: we encode repairs

ARTICLE

Repairing SHACL Constraint Violations Using Answer Set Programming

Authors: Shqiponja Ahmetaj, Robert David, Axel Polleres, Mantas Šimkus Authors Info & Claims

The Semantic Web – ISWC 2022: 21st International Semantic Web Conference, Virtual Event, October 23–27, 2022, Proceedings • Oct 2022 • Pages 375–391 • https://doi.org/10.1007/978-3-031-19433-7_22



Data Graph:

```
: Ben :enrolledIn :C1 .
:C1 :type Course.
```

vs.

```
:Ben :enrolledIn :C1.
:Ben :enrolledIn :new1.
:new1 :type Course.
```

Guess a pair (A,D) of additions and deletions that repair all target nodes

- cardinality minimality
- strategy for introducing new nodes (for minimality constaints

- relaxed encoding to repair a maximal number of target nodes
  - in the case not all target nodes can be repaired
- Implementation using Java and Clingo

SHACL … computing repairs by Guess and check:

Somewhat work in progress, since it is not 100% clear what a “good” repair semantics should look like, but – we hope - encoding this as a repair problem helps to ***clarify the semantics of the standard***:

- encode more complex repair policies
  - *e.g. fix a part of the vocabulary/signature*
- similar issues as in SPARQL++ arise when you allow recursion (cf.
- raise a discussion about intuitive repairs, may need extension of the SHACL standard!

Active work in this space, we can built upon:

RESEARCH-ARTICLE

Stable Model Semantics for Recursive SHACL

Authors: Medina Andresel, Julien Corman, Magdalena Ortiz, Juan L. Reutter, Ognjen Savkovic, Mantas Simkus Authors Info & Claims

WWW '20: Proceedings of The Web Conference 2020 • April 2020 • Pages 1570–1580 • https://doi.org/10.1145/3366423.3380229

KR Proceedings ▸ 2021 ▸ Full Papers ▸ Pages 12–21

Reasoning about Explanations for Non-validation in SHACL

Shqiponja Ahmetaj (Vienna University of Economics and Business, Austria)
Robert David (Semantic Web Company, Austria)
Magdalena Ortiz (Technical University of Vienna, Austria)
Axel Polleres (Vienna University of Economics and Business, Austria, Complexity Science Hub Vienna, Austria)
Bojken Shehu (Polytechnic University of Tirana, Albania)
Mantas Šimkus (Technical University of Vienna, Austria)

Why I love ASP (since over 20 years)…

- ***Intuitive***, understandable Problem ***encodings***…`
- … easily ***extensible***
- the beauty of **Guess and Check** to solve complex problems on top

Part 1:
1999-2003

AI Planning

Part 2:
2003 – to date…

Universidad
Rey Juan Carlos

Semantic Web

Part 3:
ca. 2014 – 2022…

Business Process
Management

Resource allocation in BPM

- Recently concluded PhD thesis:

Giray Havur

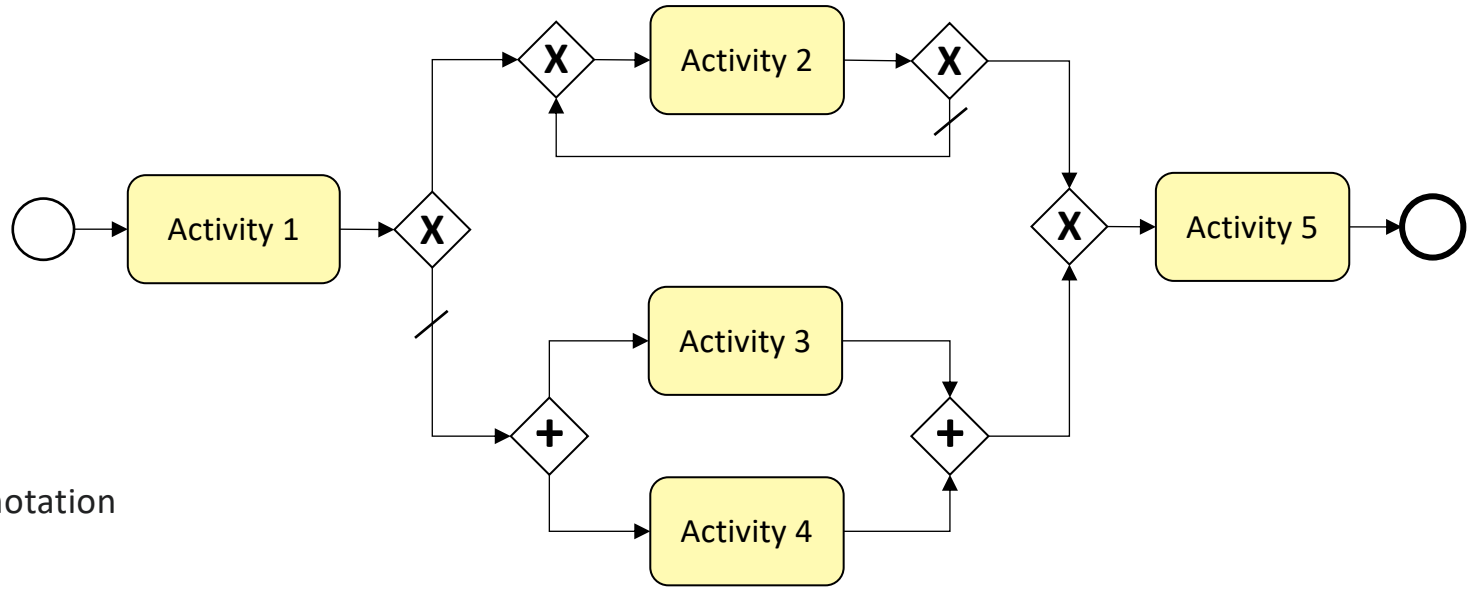
SHAPE

Safety-critical Human- and dAta-centric Process management in Engineering projects

SHAPE

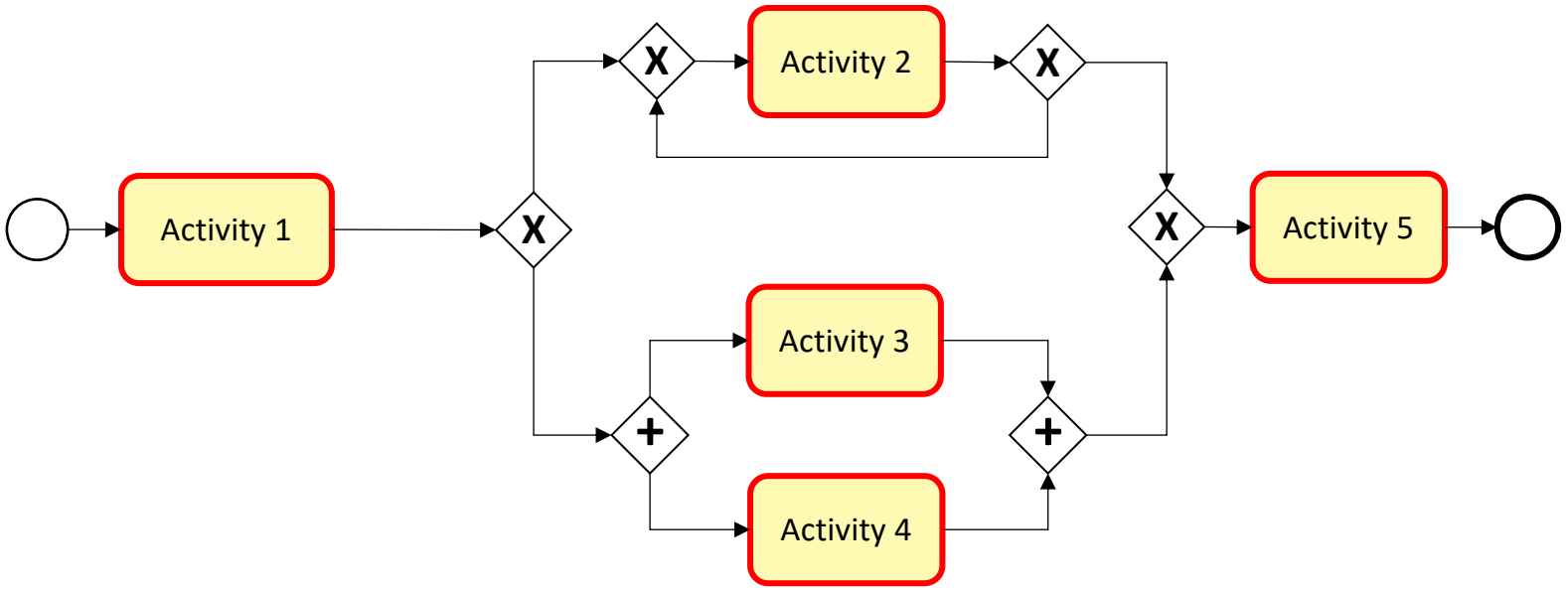
Co-supervisor/Co-PI

Organizing Work
Business Process Modelling Notation

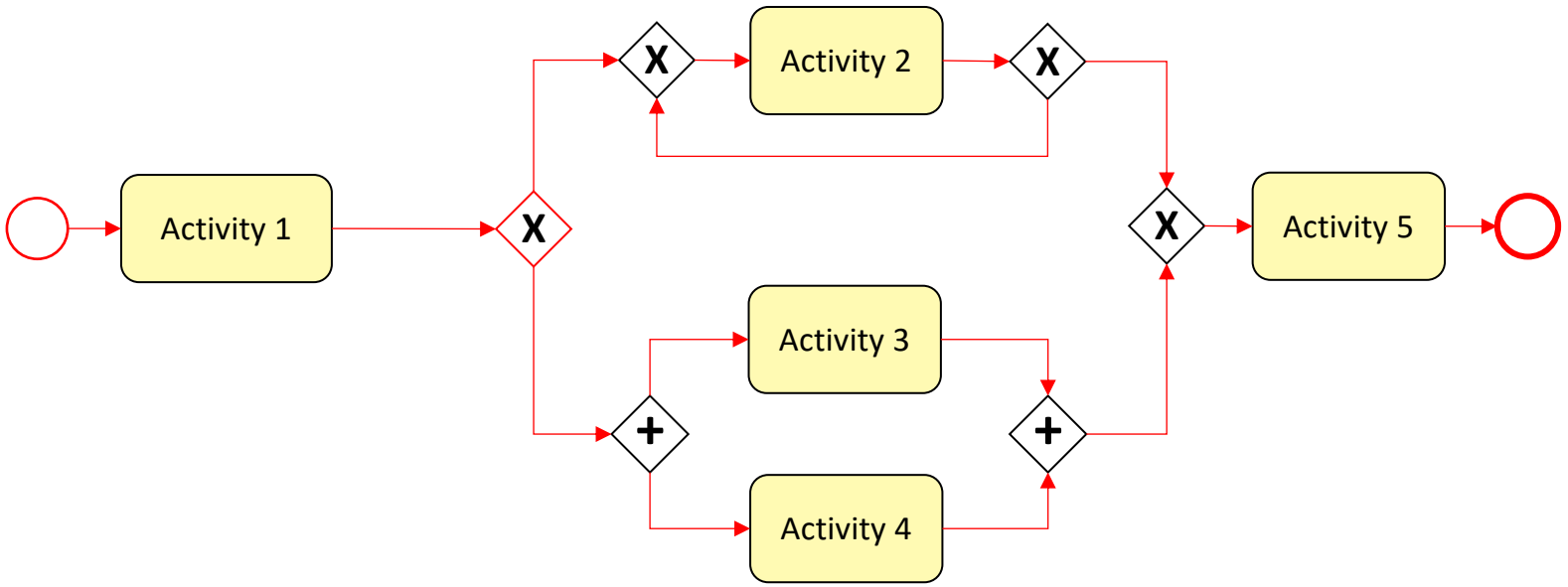


- A standardized and popular graphical notation for specifying business processes

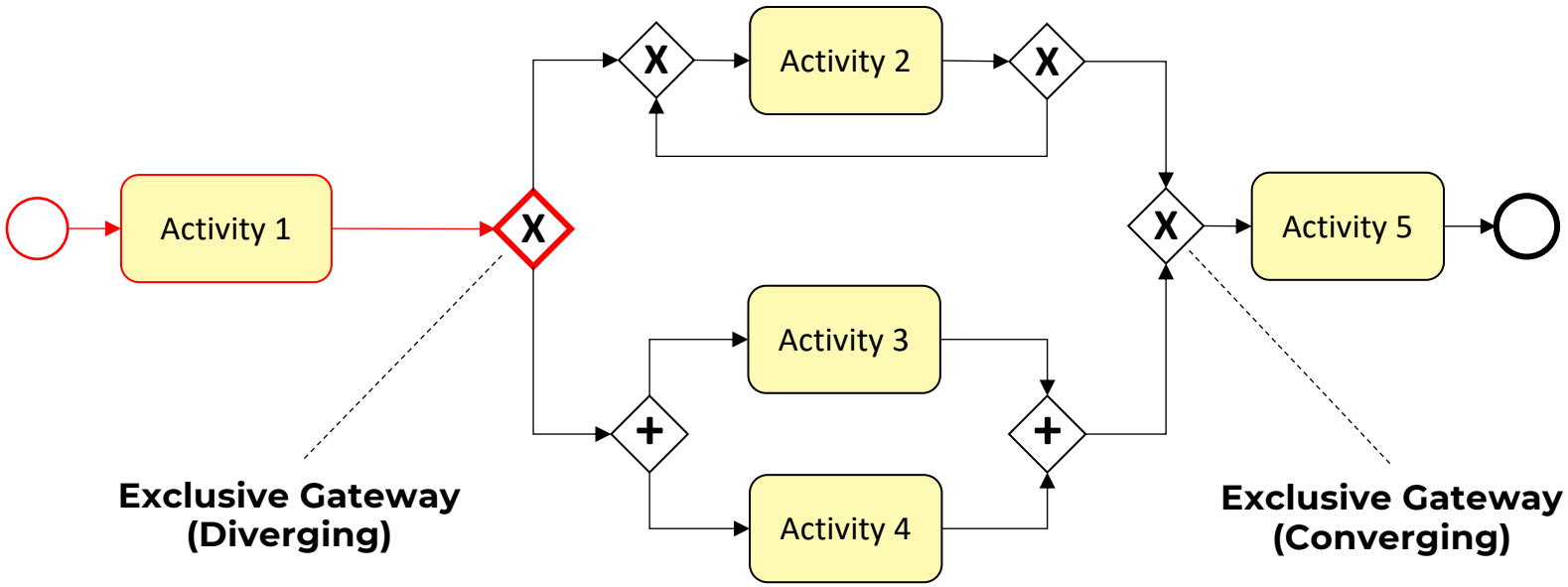
Organizing Work
Business Process Modelling Notation



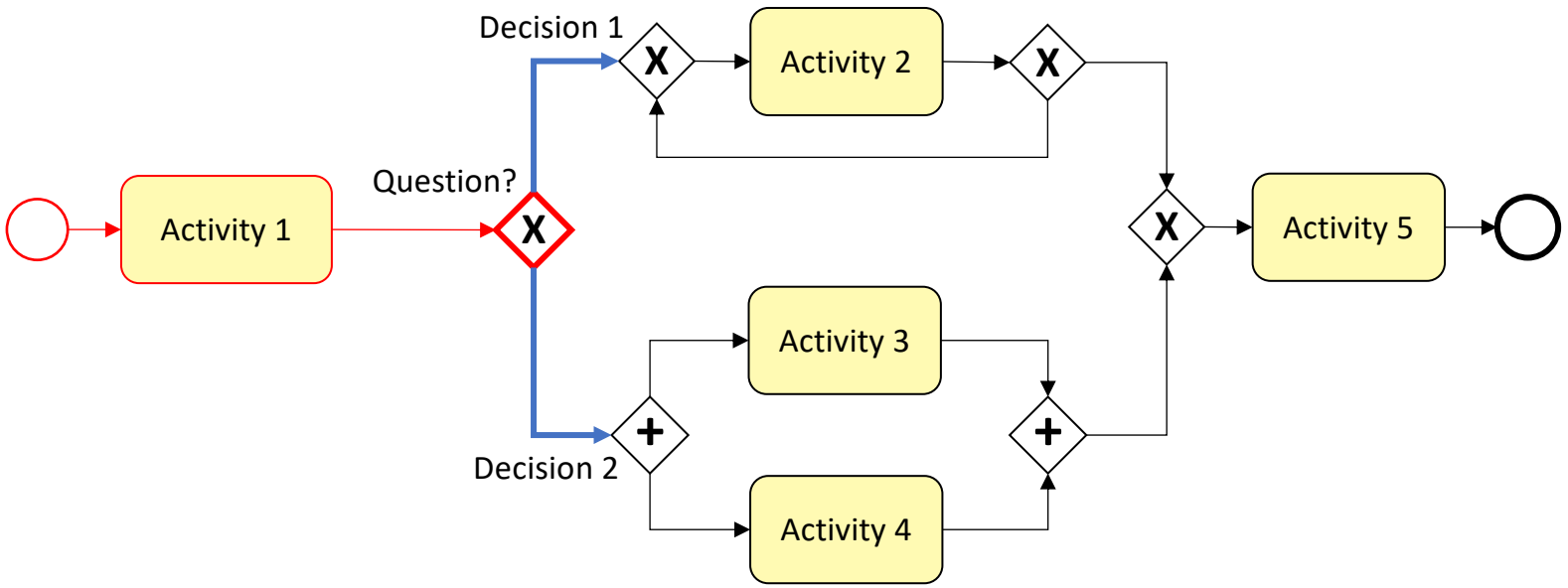
Organizing Work
Business Process Modelling Notation



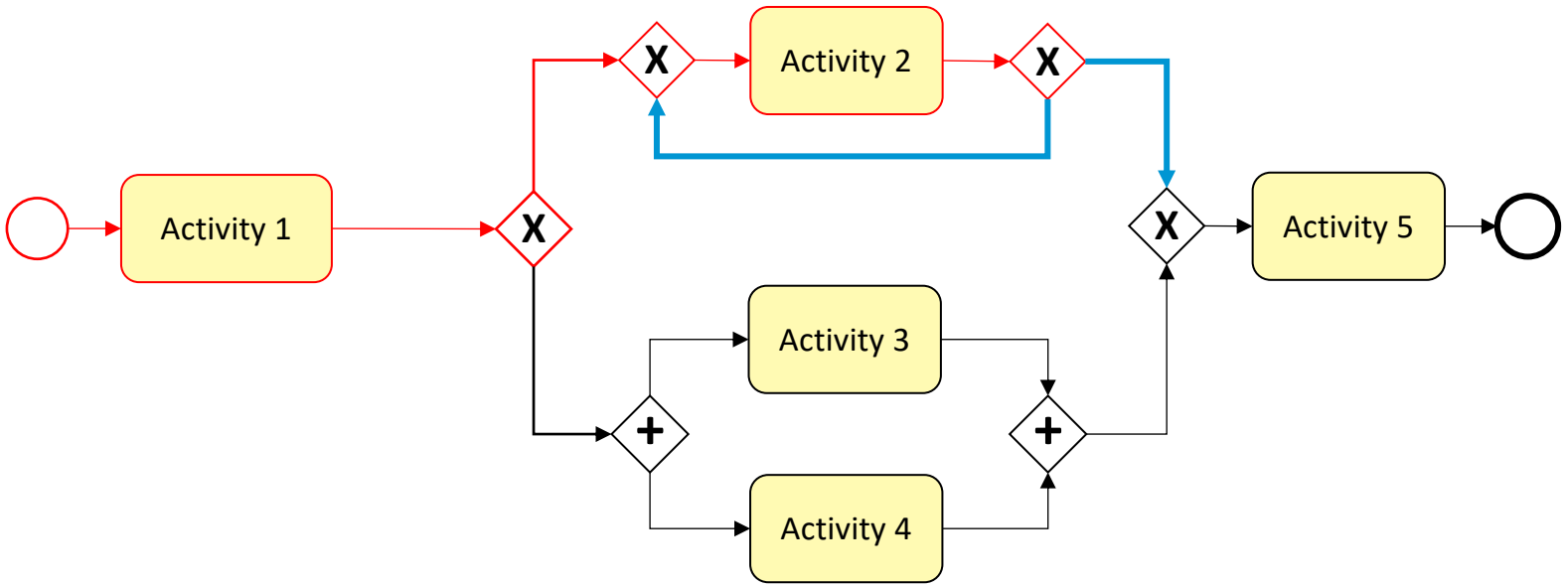
Organizing Work
Business Process Modelling Notation



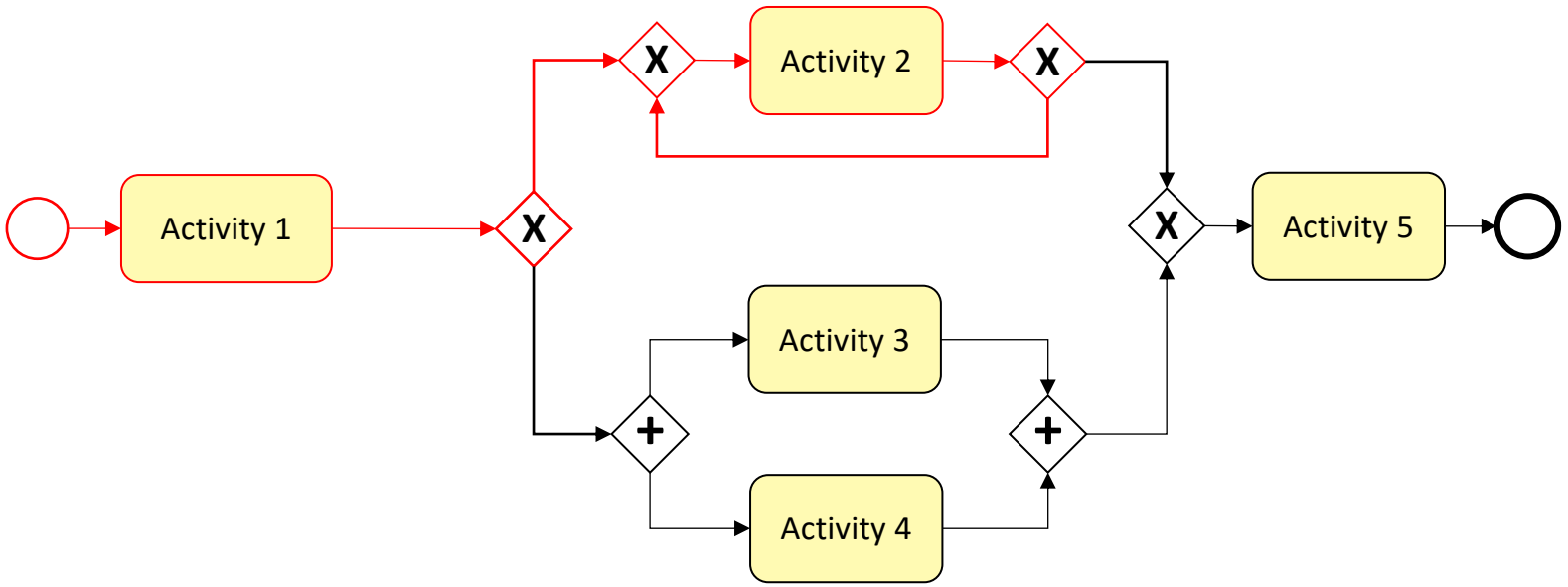
Organizing Work
Business Process Modelling Notation



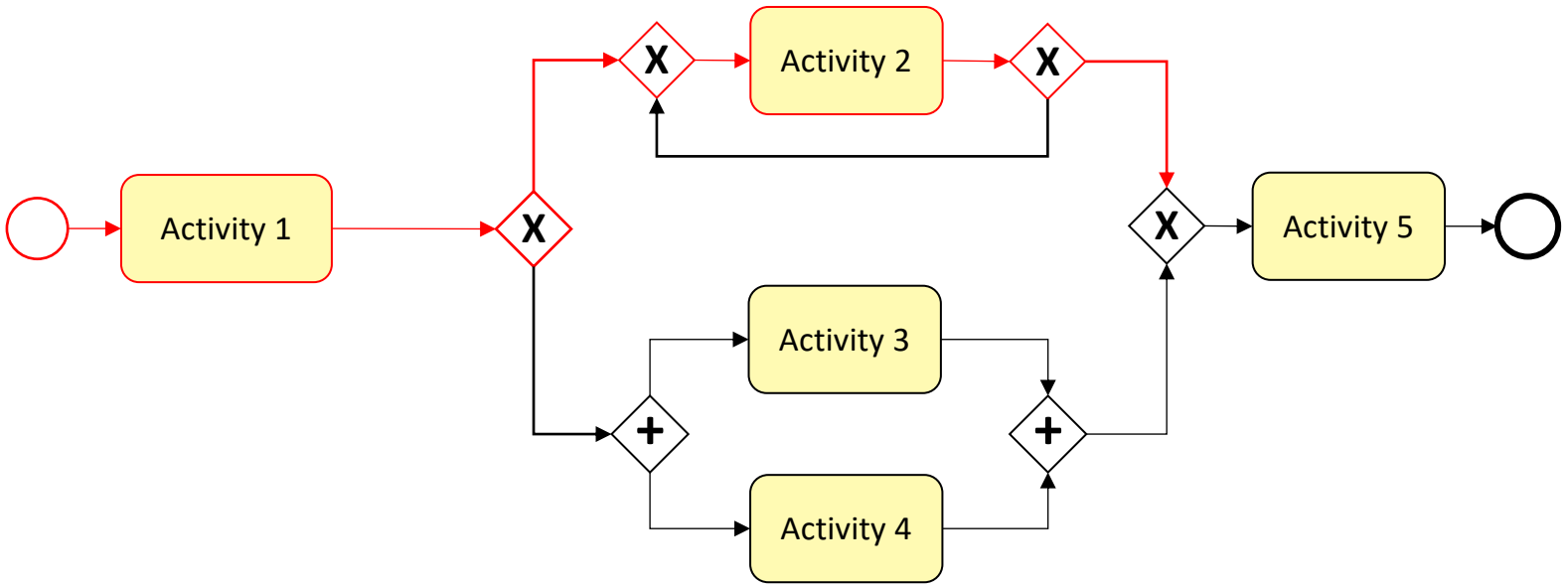
Organizing Work
Business Process Modelling Notation



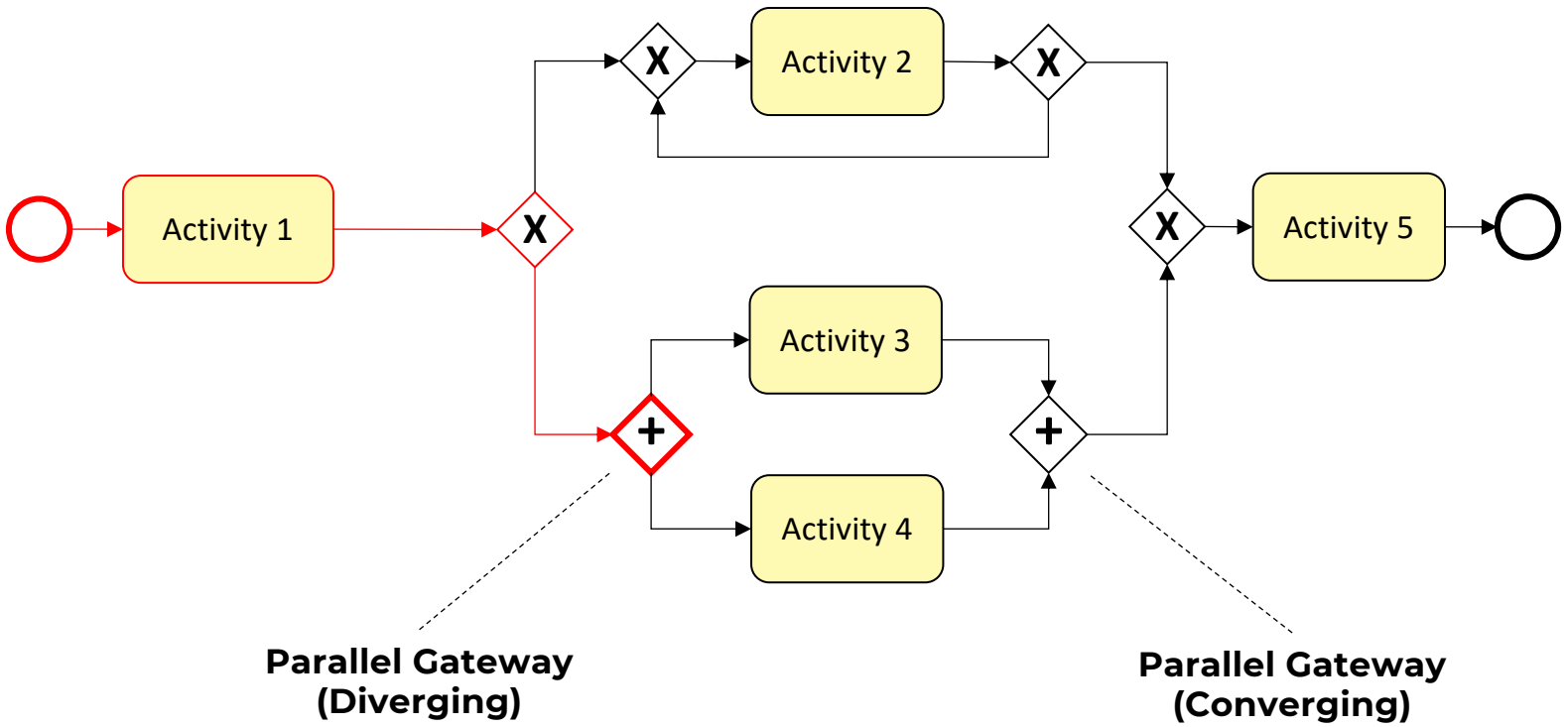
Organizing Work
Business Process Modelling Notation



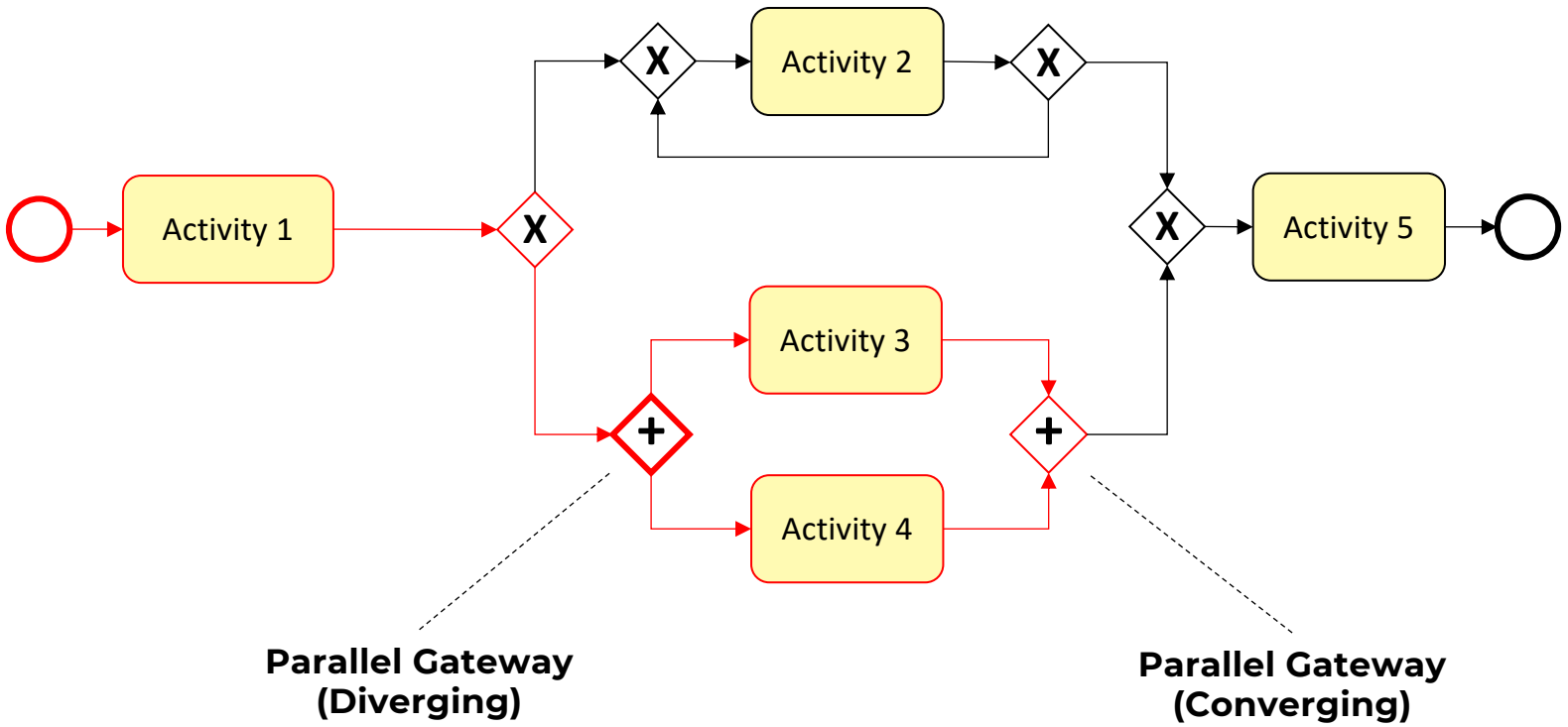
Organizing Work
Business Process Modelling Notation



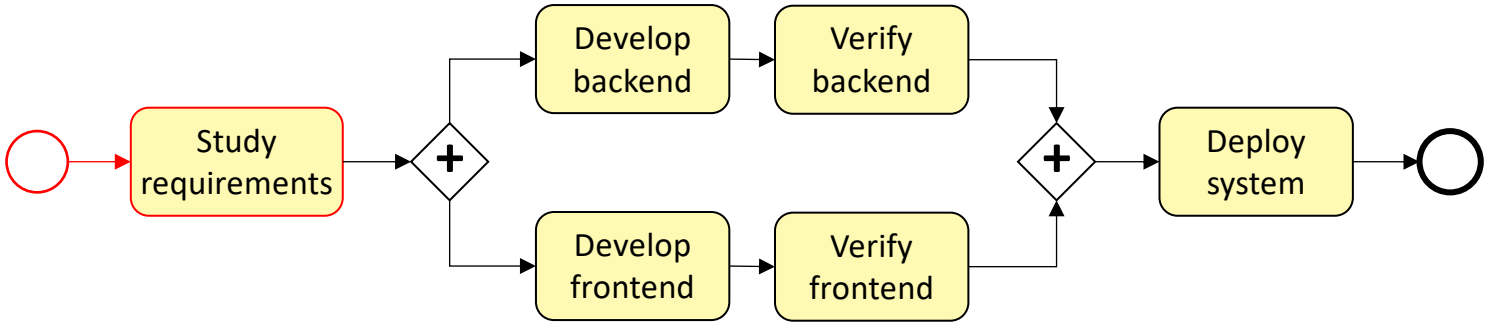
Organizing Work
Business Process Modelling Notation



Organizing Work
Business Process Modelling Notation

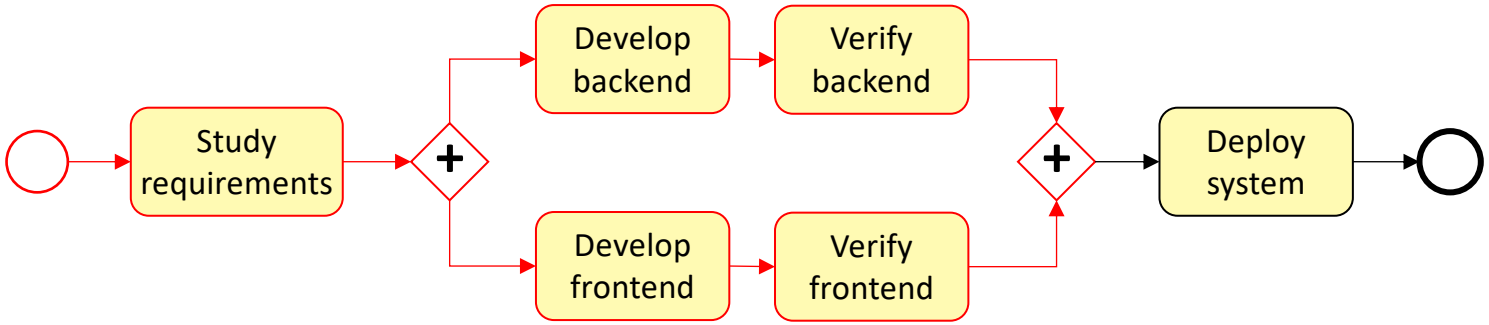


Organizing Work
BPMN Example



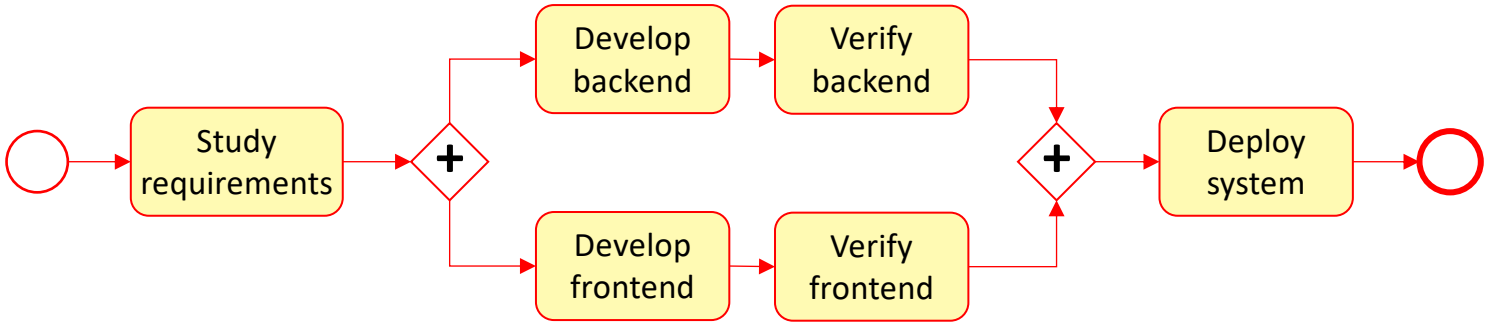
Process Model: A BPMN Model for Software Development

Organizing Work
BPMN Example



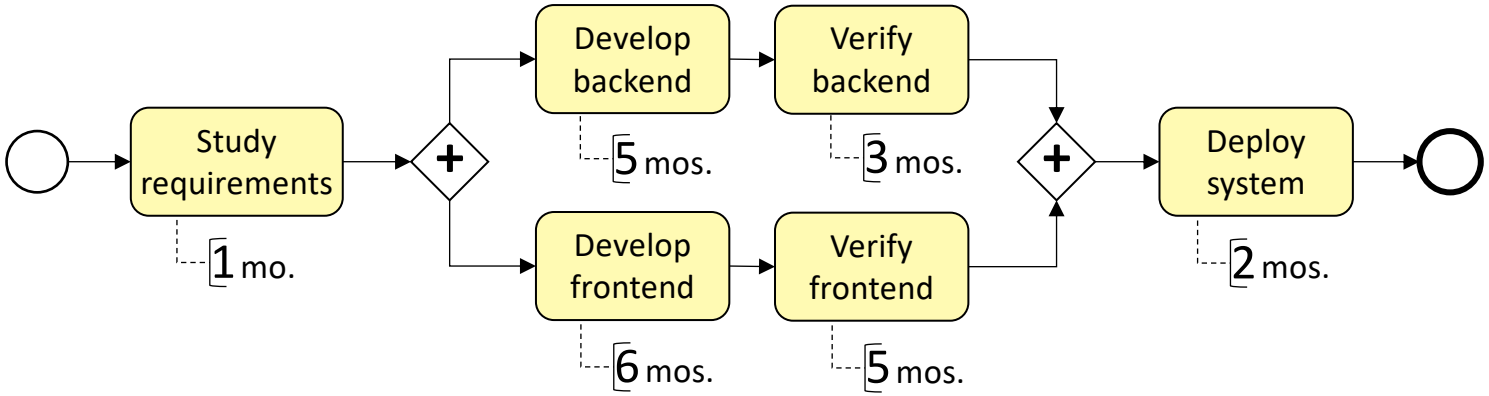
Process Model: A BPMN Model for Software Development

Organizing Work
BPMN Example



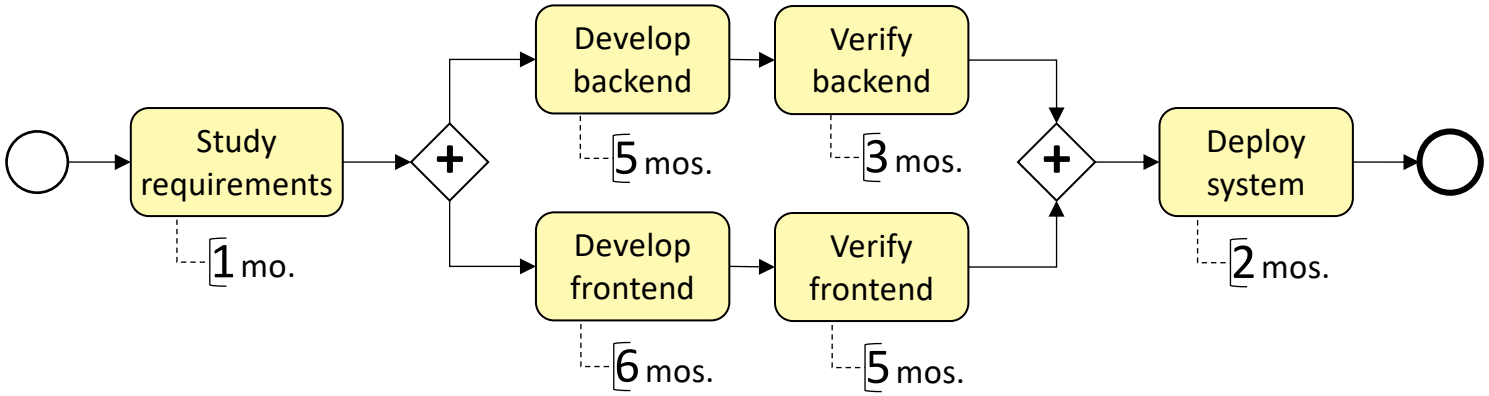
Process Model: A BPMN Model for Software Development

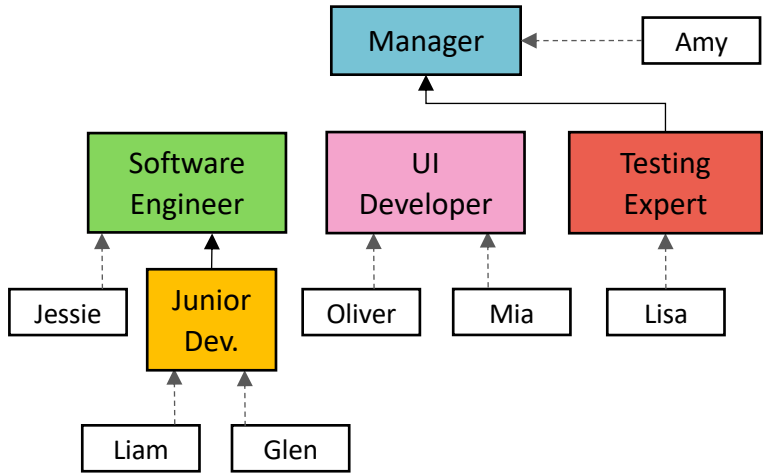
Organizing Work
BPMN Example



Process Model: A BPMN Model for Software Development

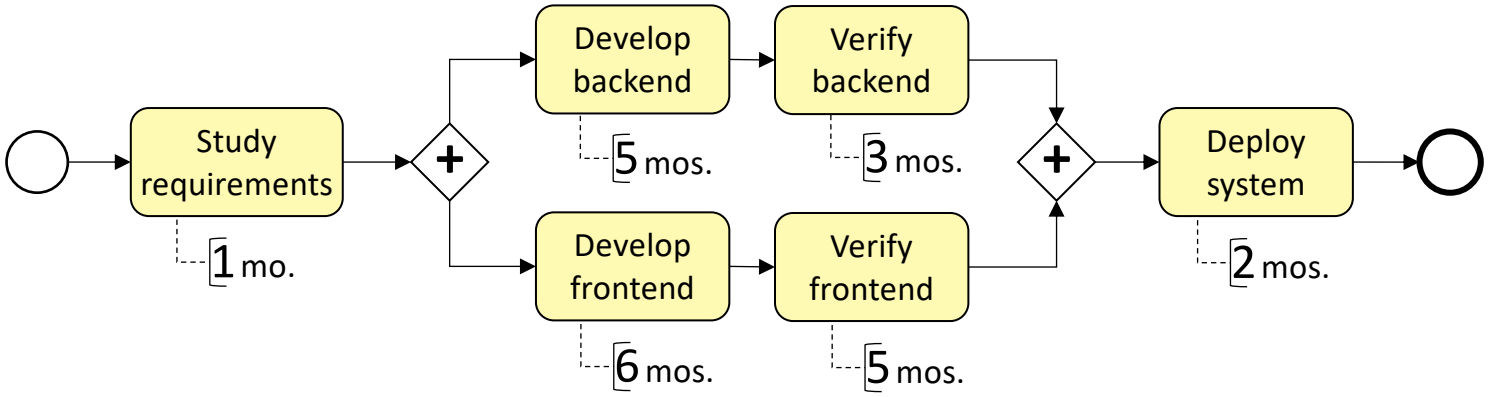
Organizing Resources
Role-based Access Control Model

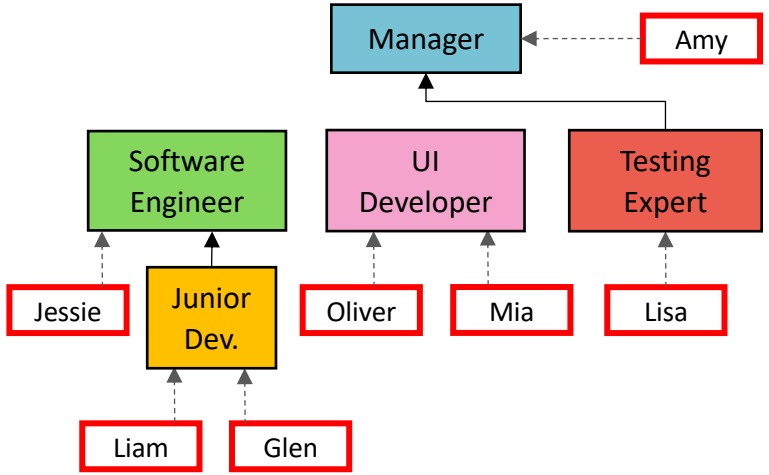




Organizational Model: An RBAC Model of the Software Development Company

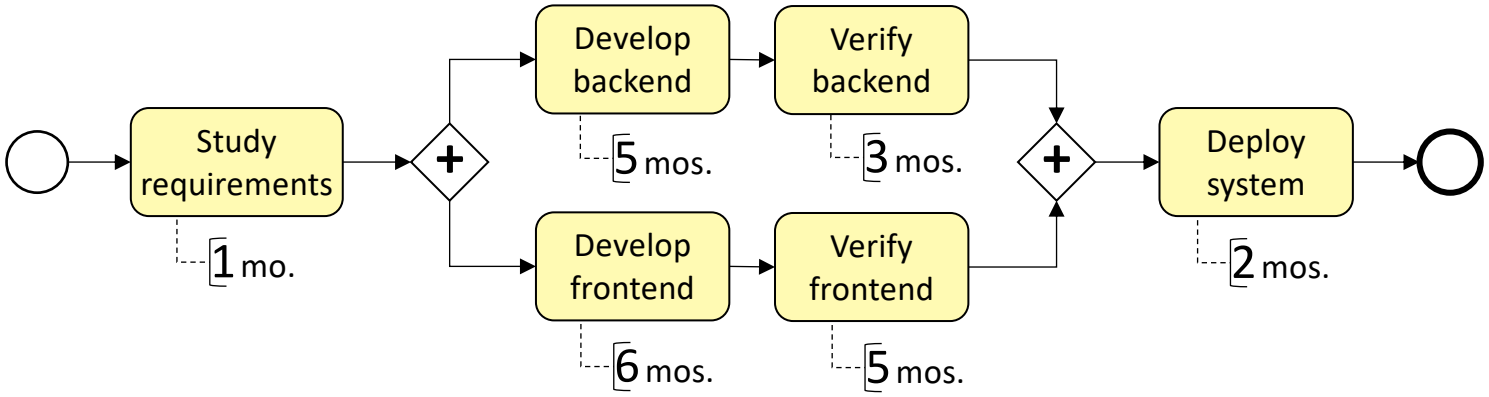
Organizing Resources
Role-based Access Control Model

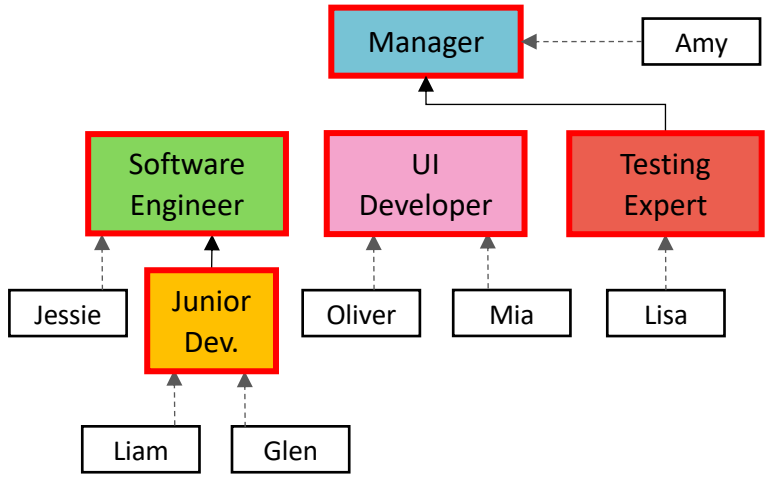




- **Resources**

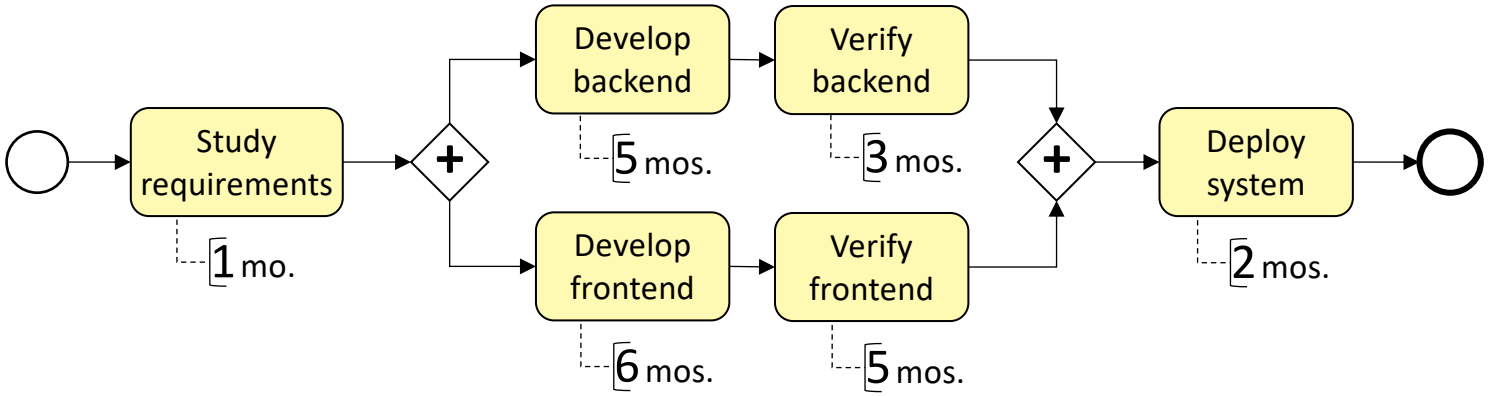
Organizing Resources
Role-based Access Control Model

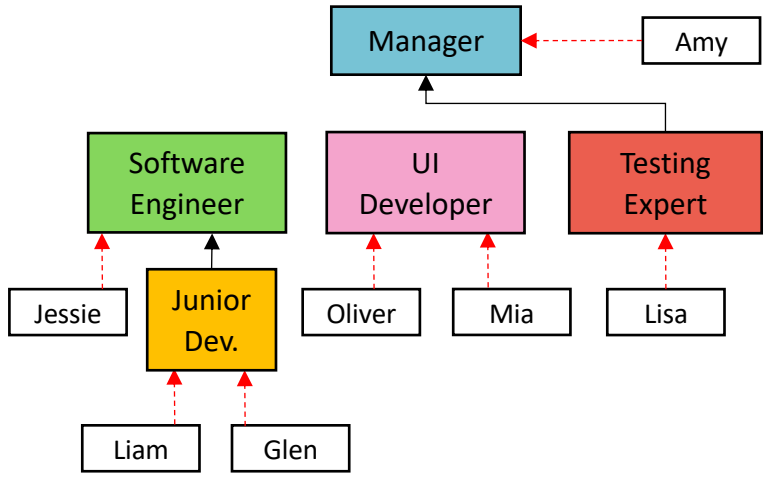




- Resources
- **Roles**

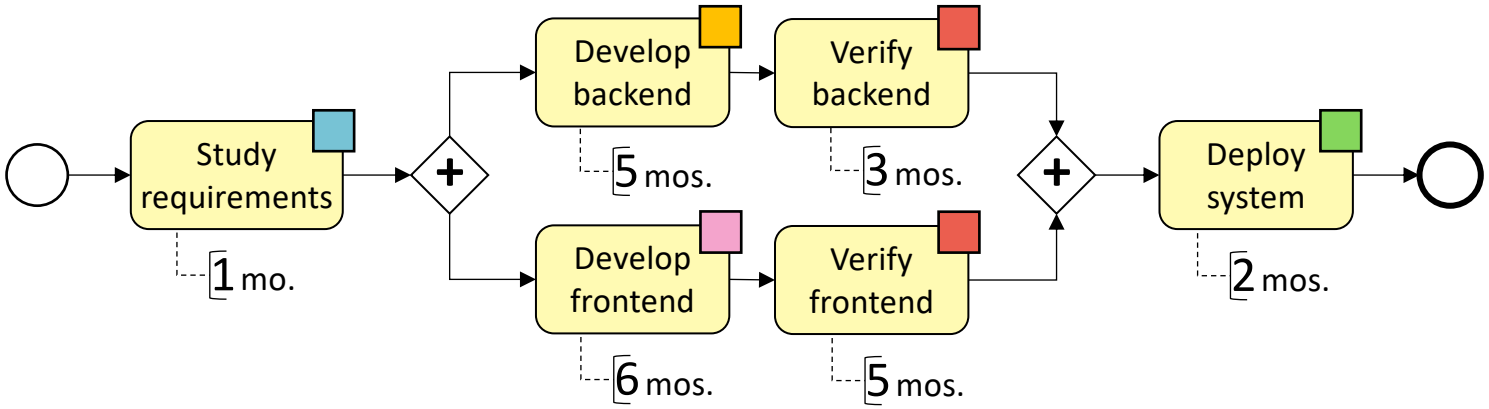
Organizing Resources
Role-based Access Control Model

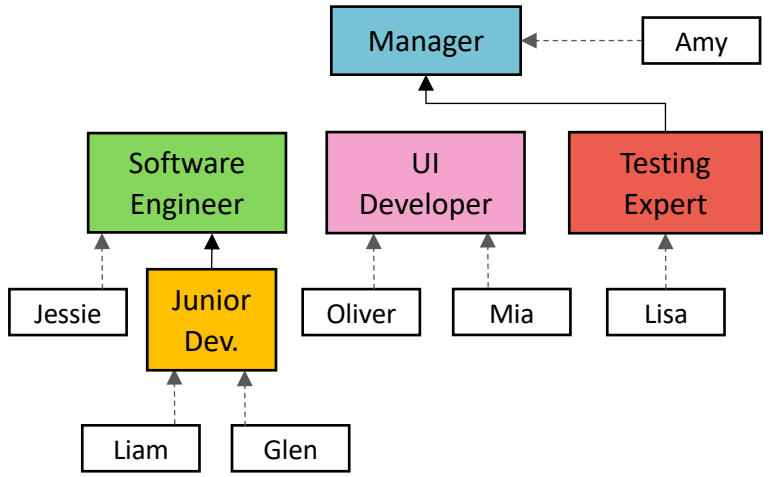




- Resources
- Roles
- **Resource-to-Role assignments**

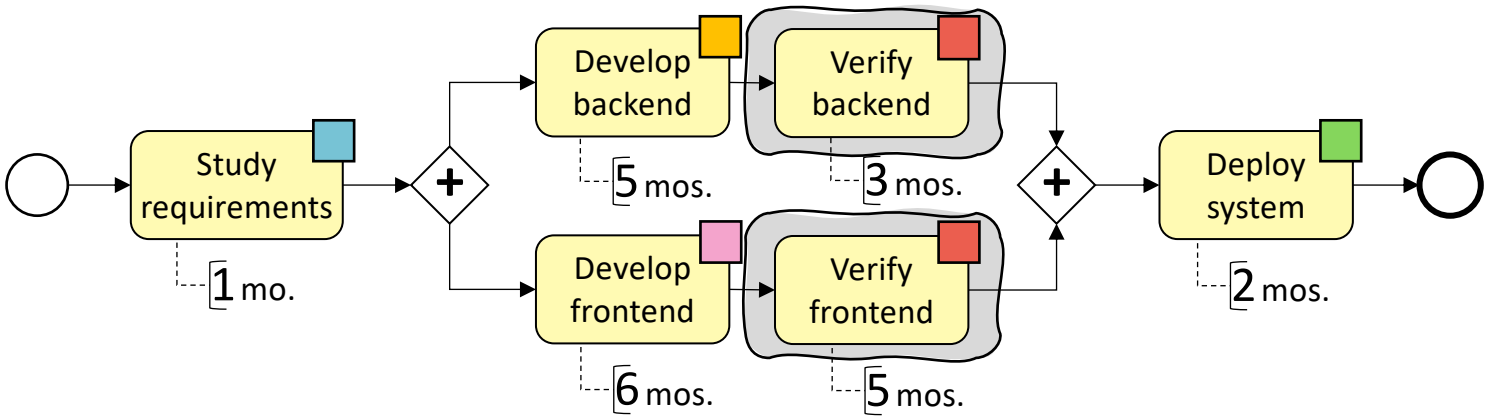
Organizing Resources
Role-based Access Control Model

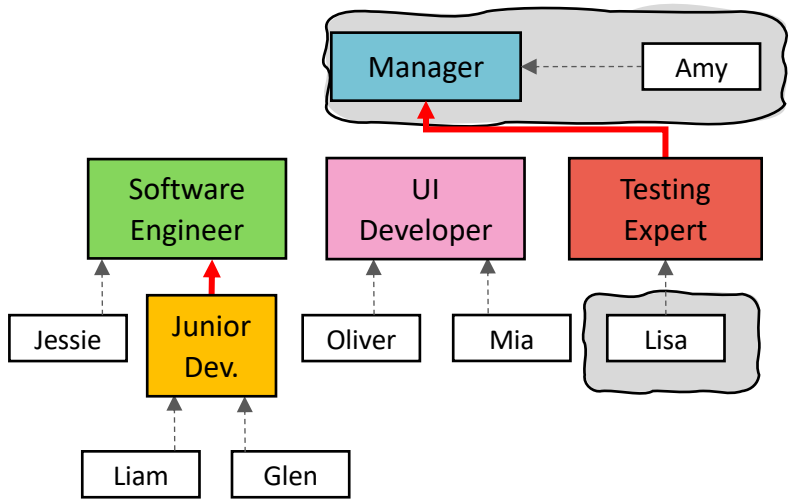




- Resources
- Roles
- Resource-to-Role assignments
- **Activity-to-Role assignments**

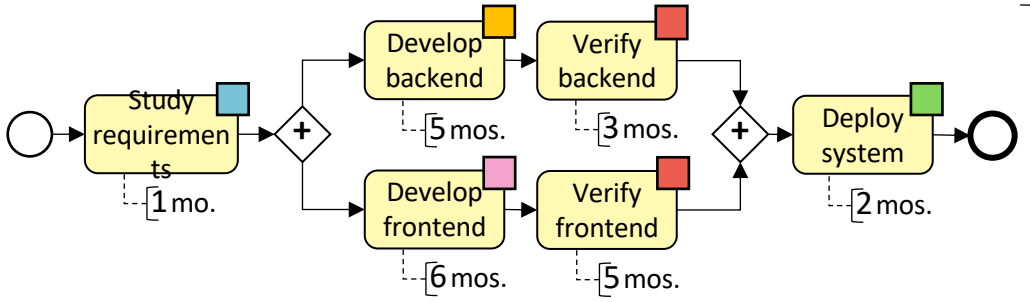
Organizing Resources
Role-based Access Control Model

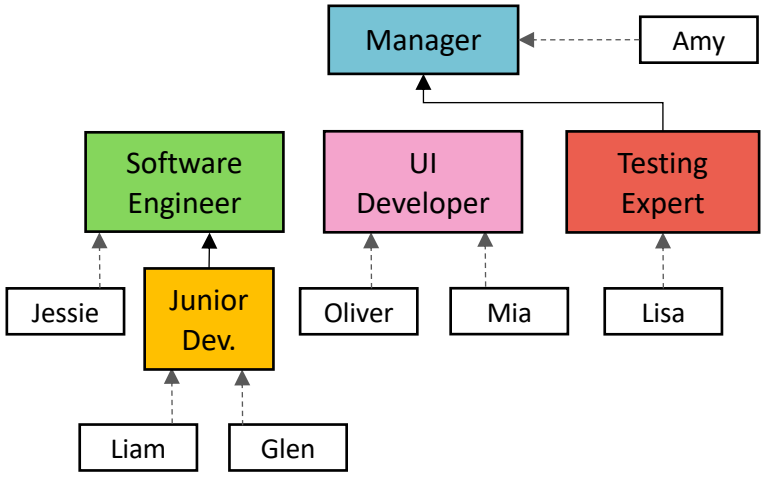


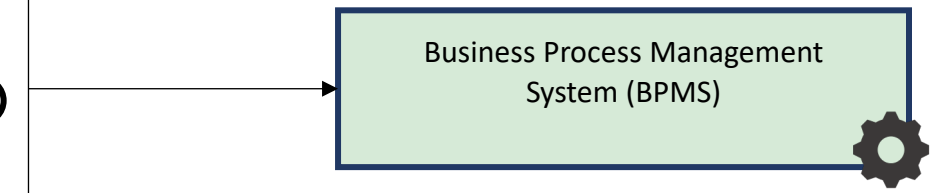


- Resources
- Roles
- Resource-to-Role assignments
- Activity-to-Role assignments
- **Role-to-Role assignments**

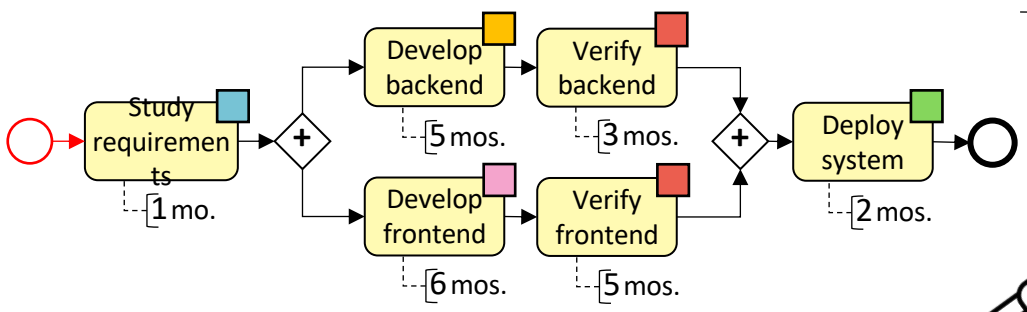
Running Business Processes
Business Process Management System

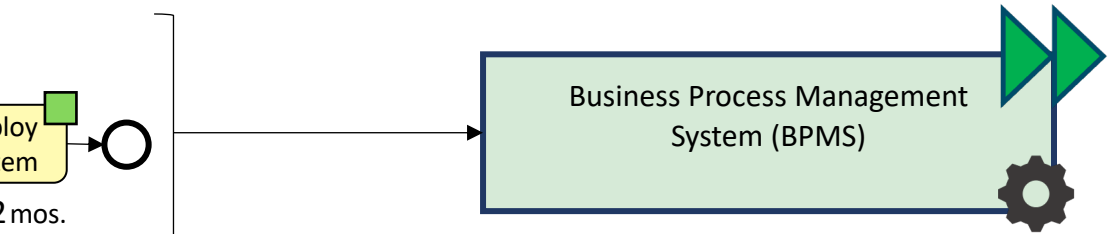


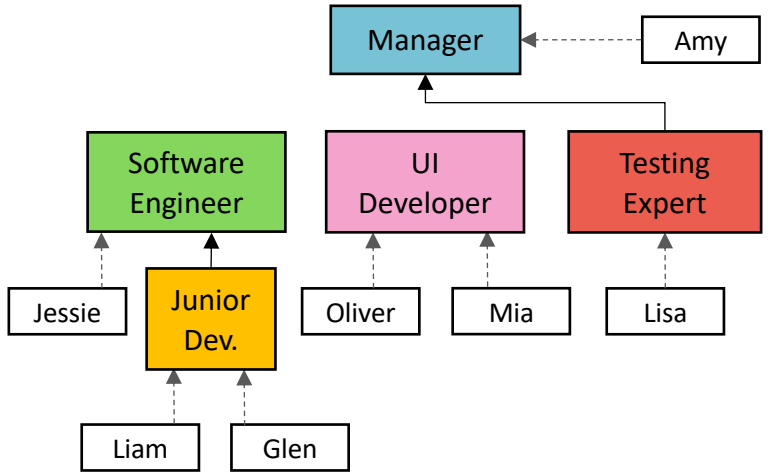




Running Business Processes
Business Process Management System





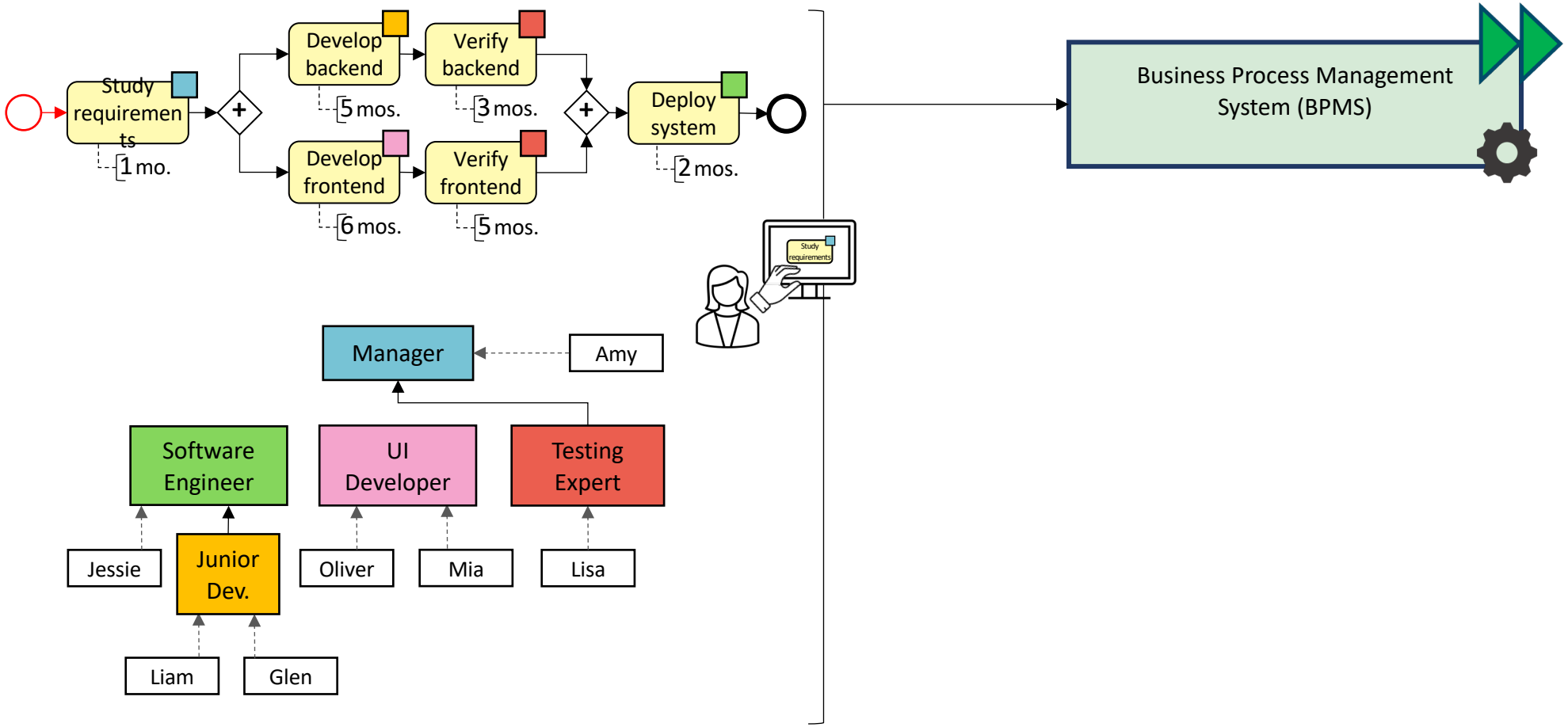


Typical approaches for BPM systems:

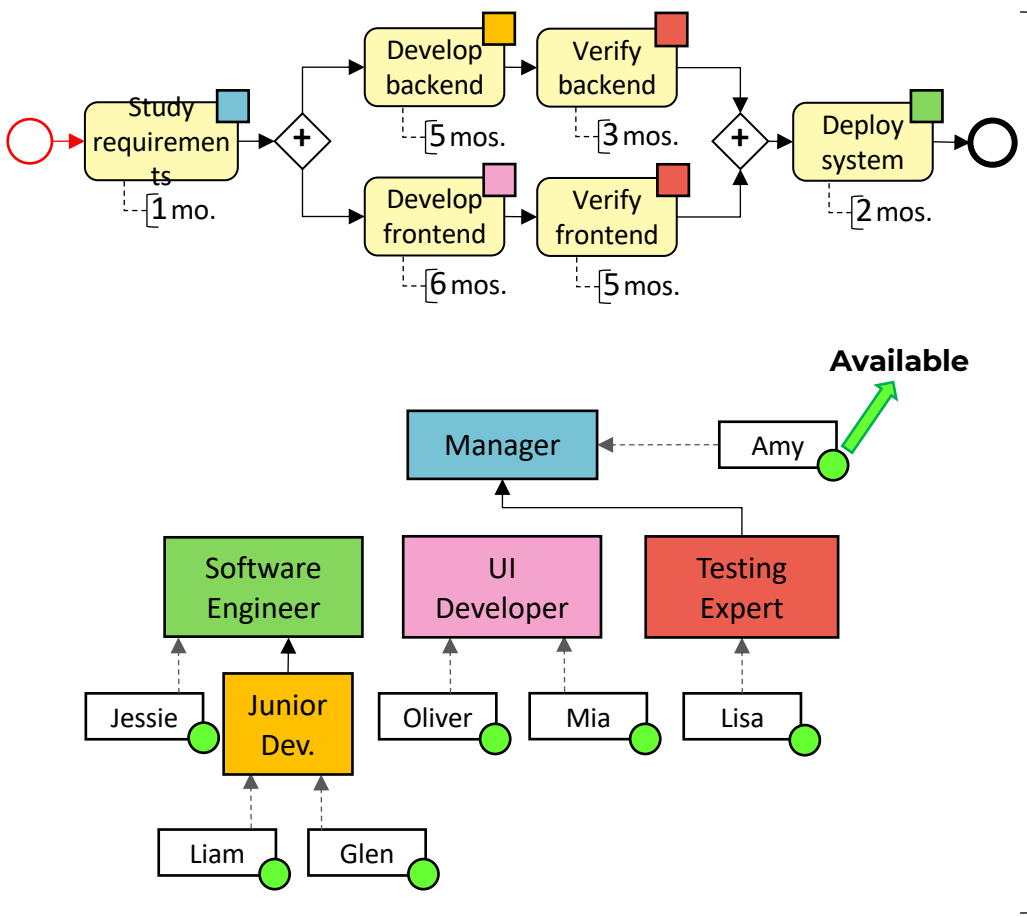
- pull-based or random allocation,
- allocation-by-order

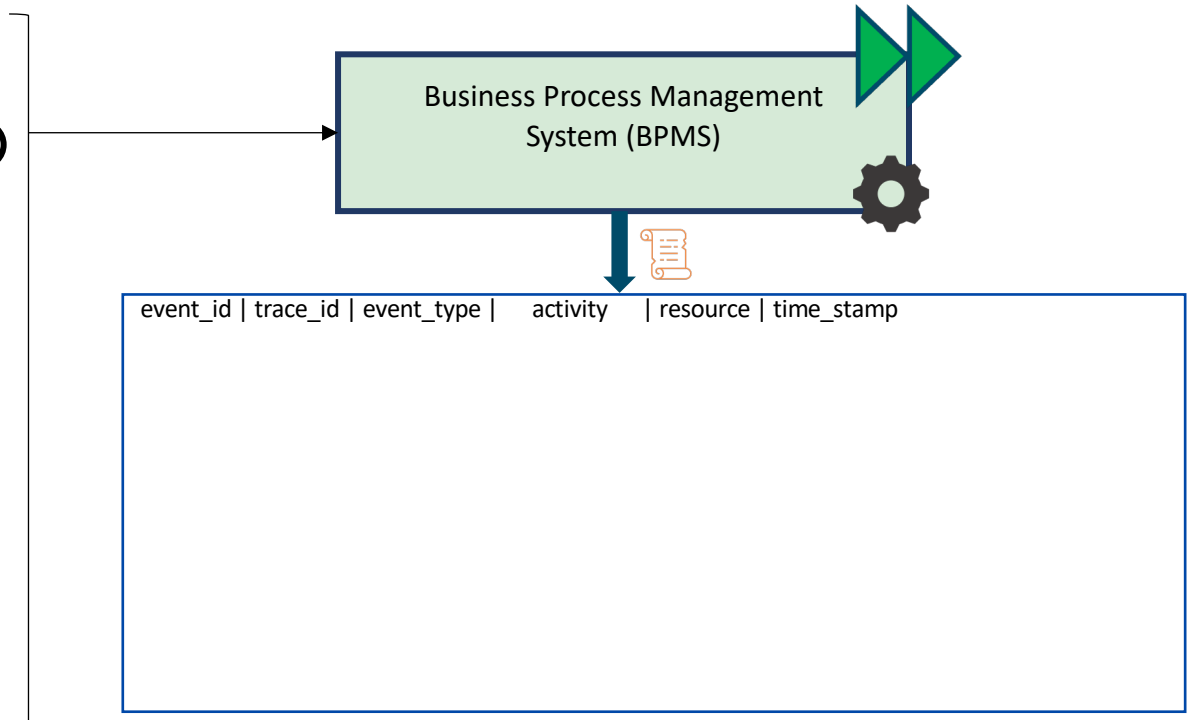
but ***scheduling*** aspect not considered so far, rather: allocation by process managers.

Running Business Processes
Business Process Management System

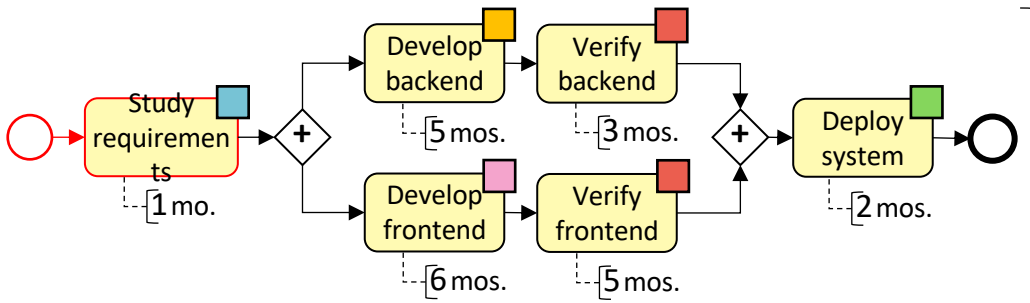


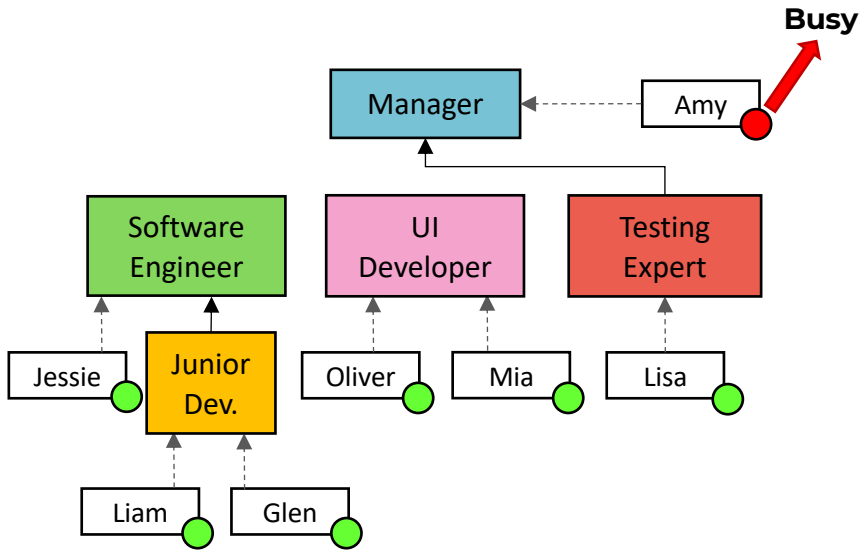
Running Business Processes
Business Process Management System

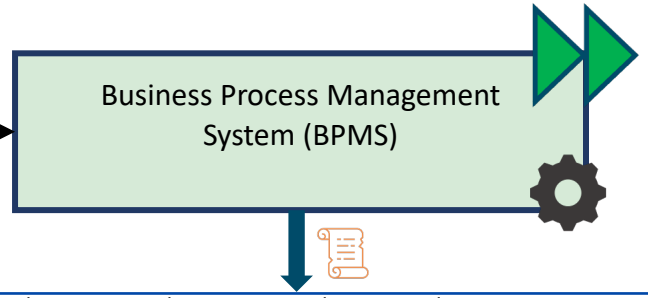




Running Business Processes
Business Process Management System

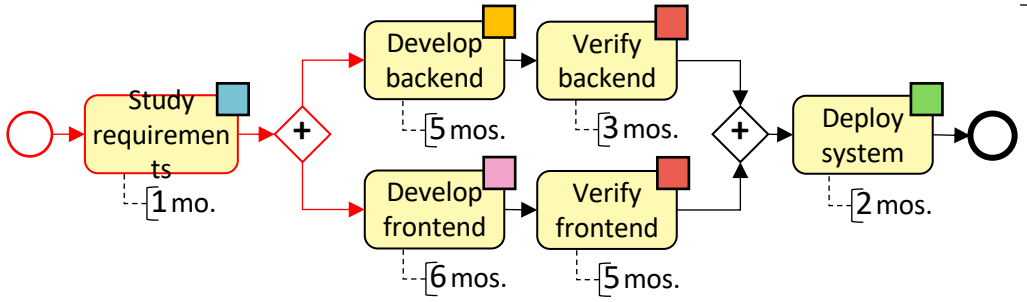


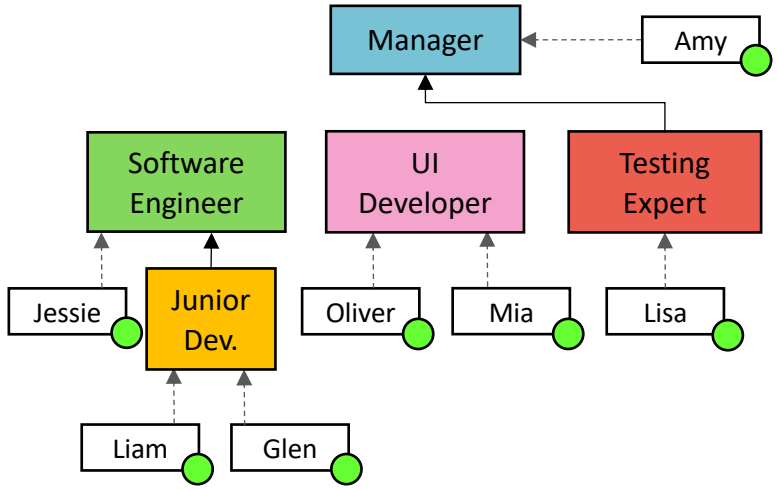


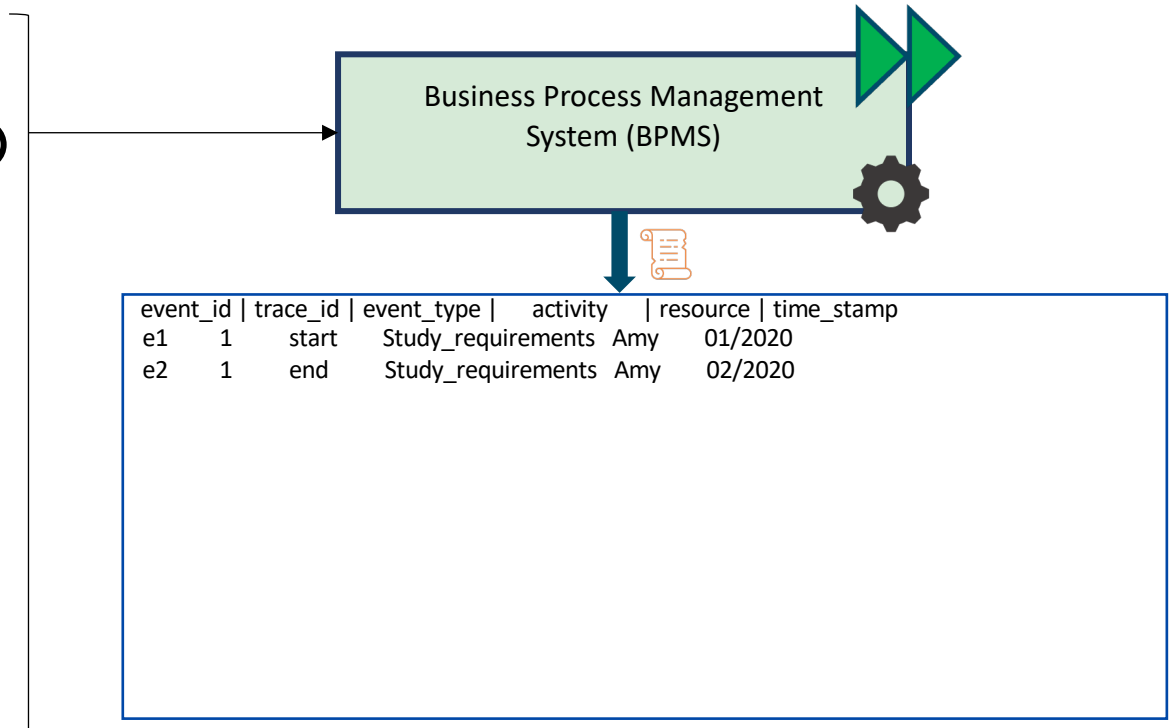


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020

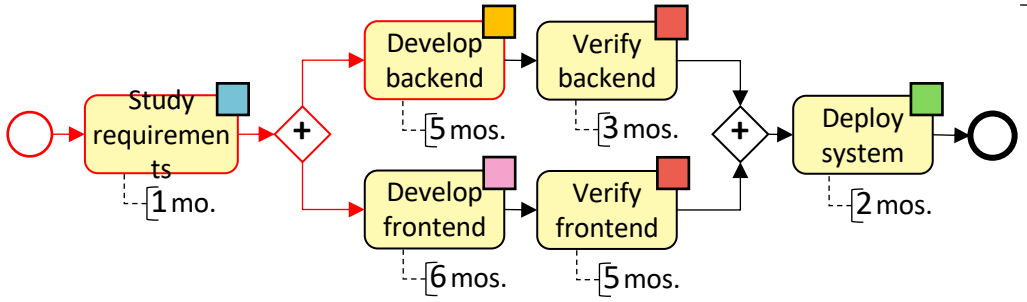
Running Business Processes
Business Process Management System

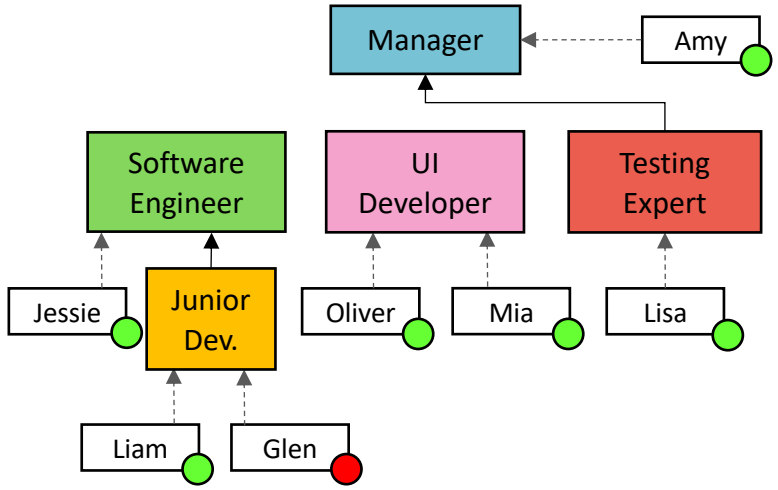


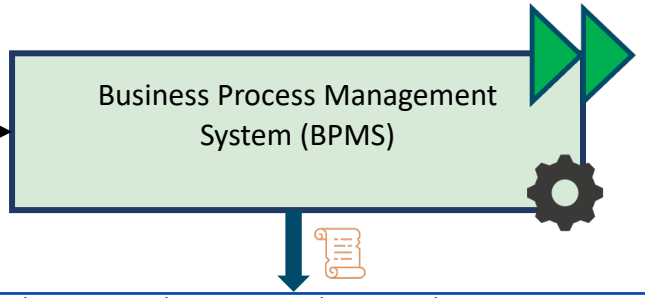




Running Business Processes
Business Process Management System

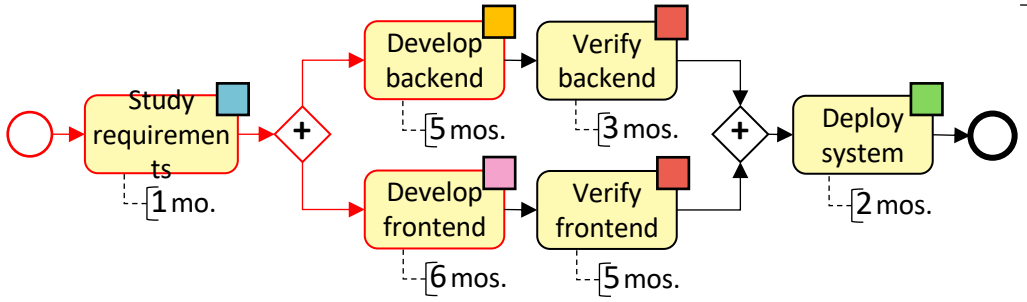


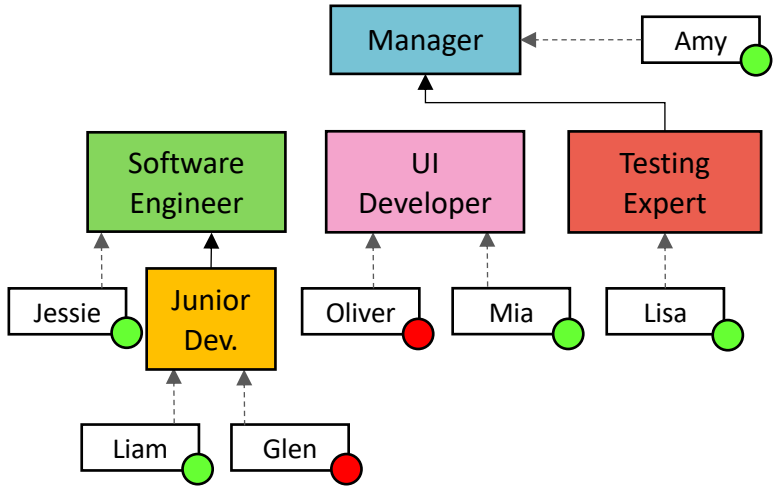


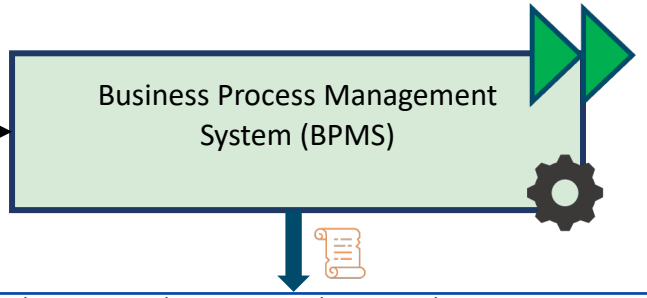


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020

Running Business Processes
Business Process Management System

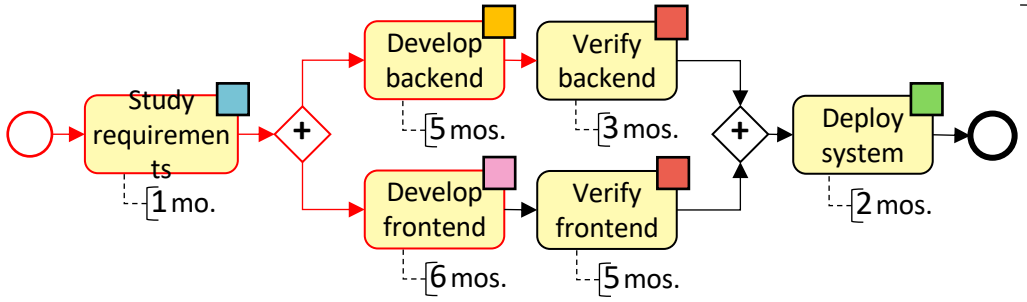


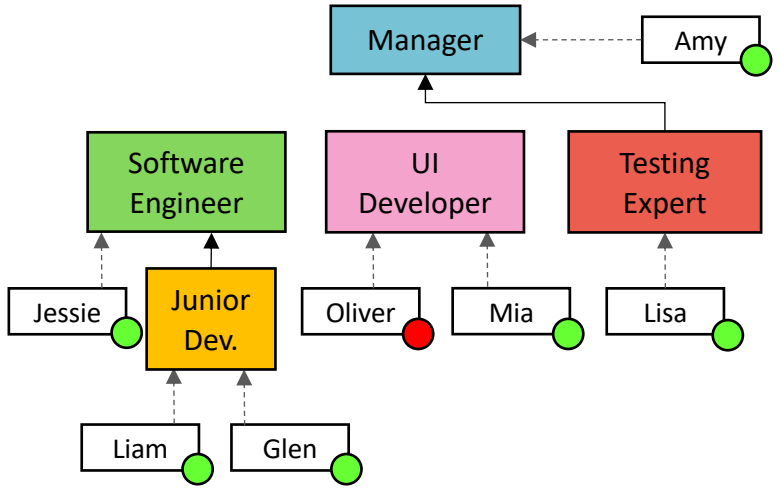


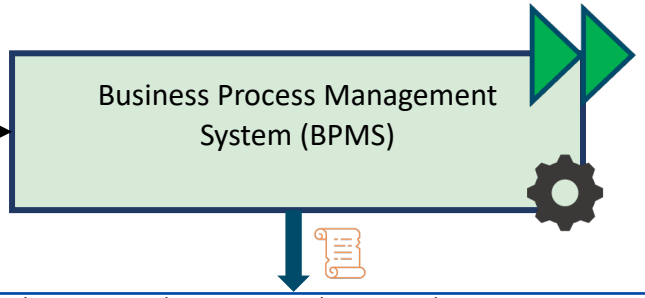


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020

Running Business Processes
Business Process Management System

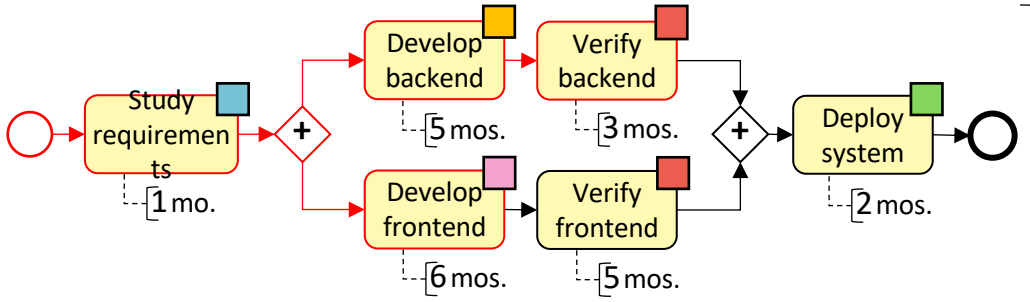


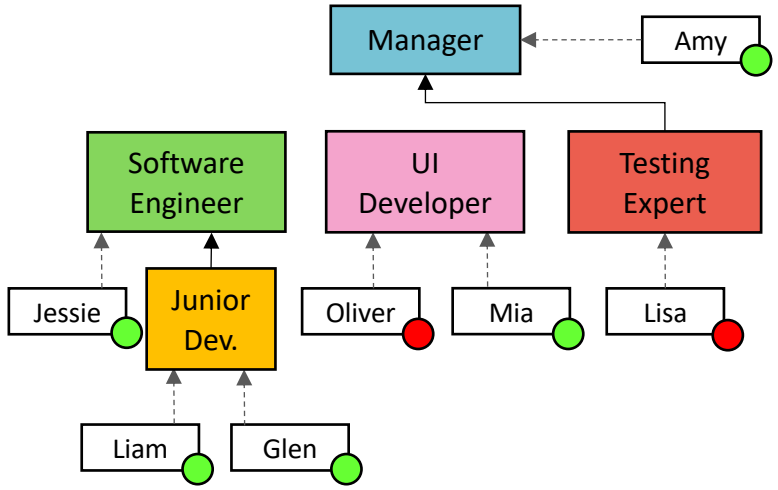


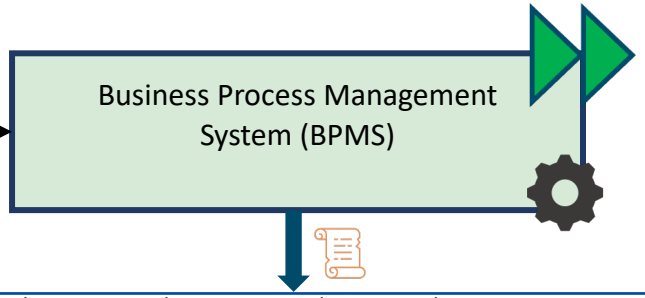


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020

Running Business Processes
Business Process Management System

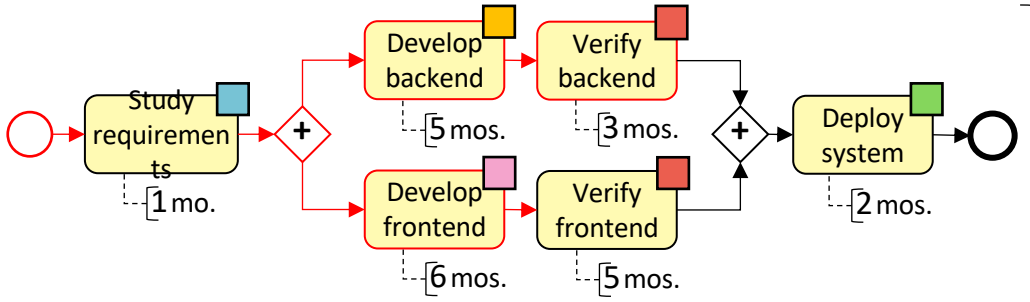


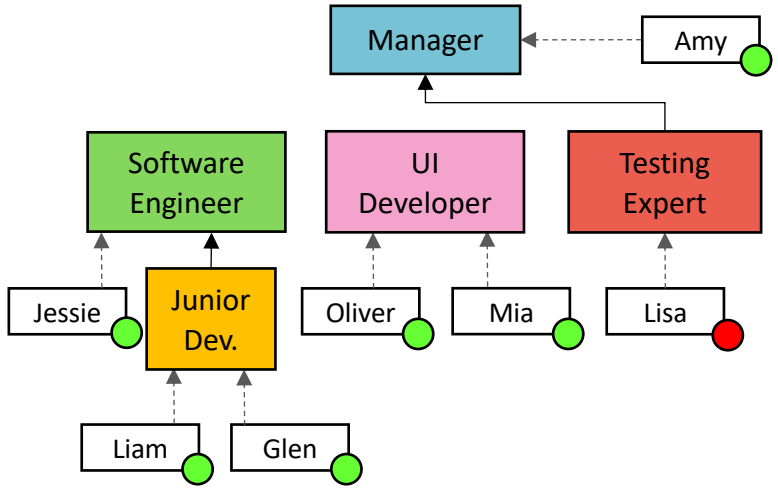


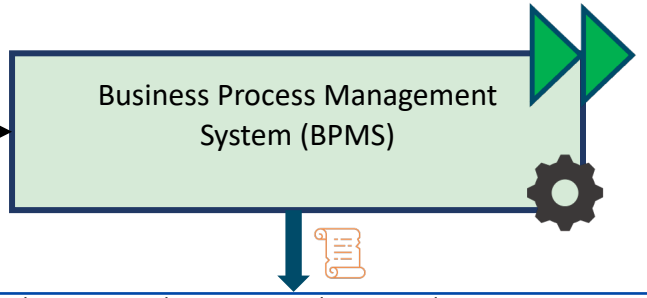


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020
e6	1	start	Verify_backend	Lisa	07/2020

Running Business Processes
Business Process Management System

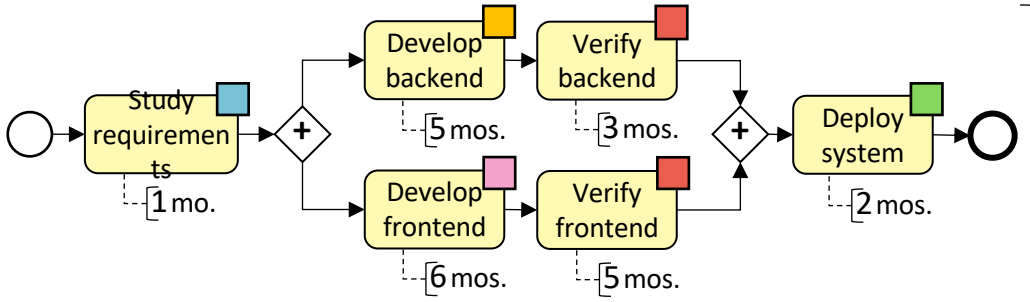


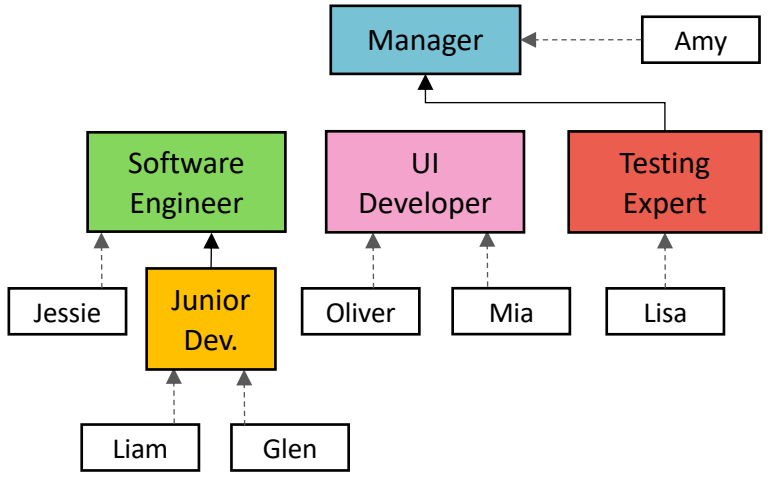


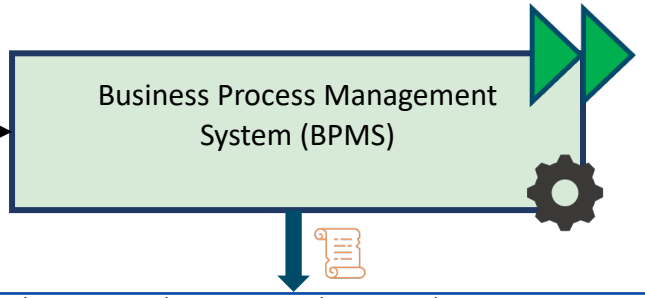


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020
e6	1	start	Verify_backend	Lisa	07/2020
e7	1	end	Develop_frontend	Oliver	10/2020

Running Business Processes
Business Process Management System

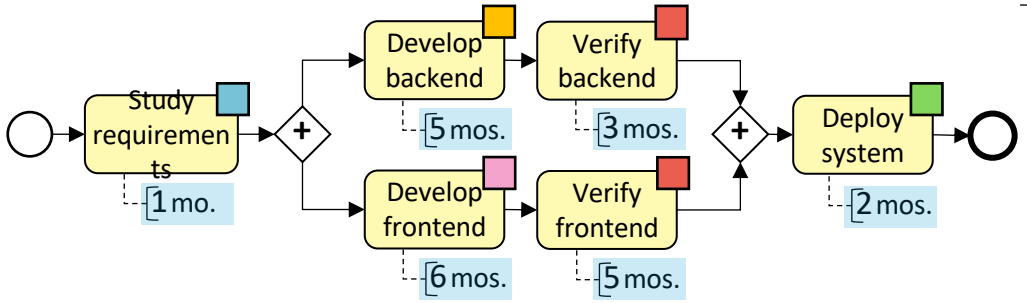


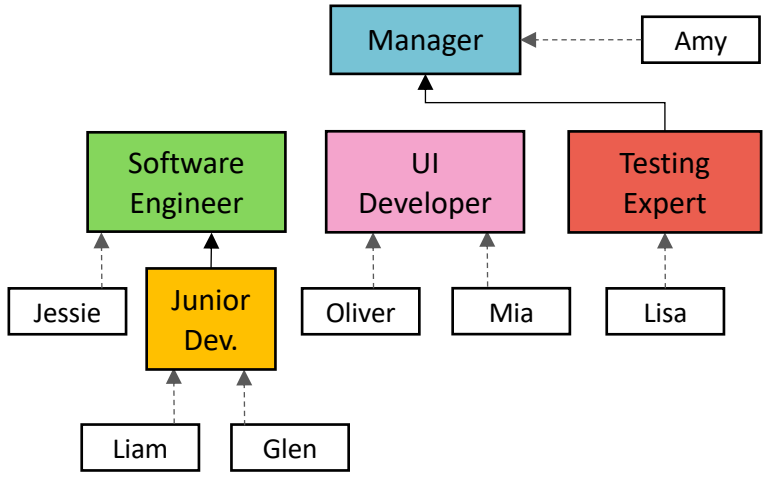


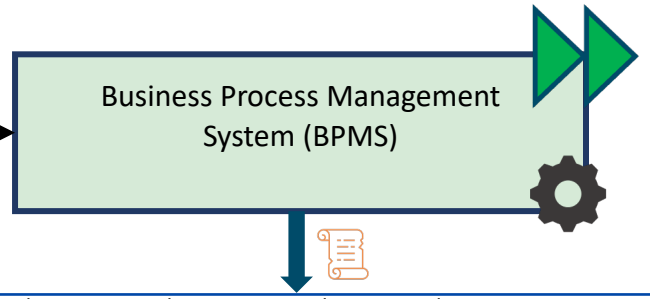


event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020
e6	1	start	Verify_backend	Lisa	07/2020
e7	1	end	Develop_frontend	Oliver	10/2020
e8	1	end	Verify_backend	Lisa	10/2020
e9	1	start	Verify_frontend	Lisa	10/2020
e10	1	end	Verify_frontend	Lisa	11/2020
e11	1	start	Deploy_system	Jessie	11/2020
e12	1	end	Deploy_system	Jessie	01/2021

Running Business Processes
Mining a Temporal Model from the Event Log



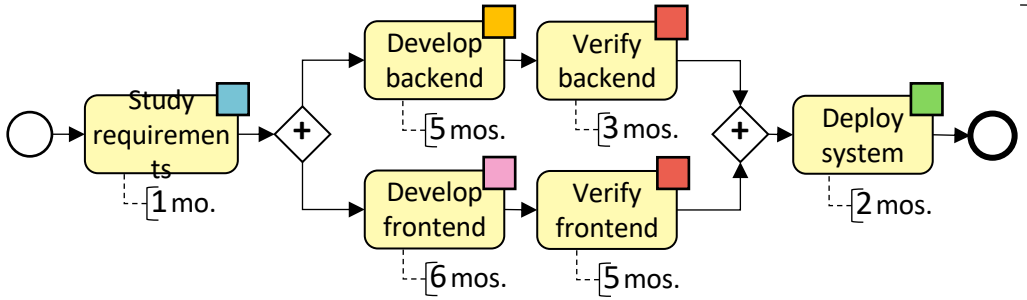


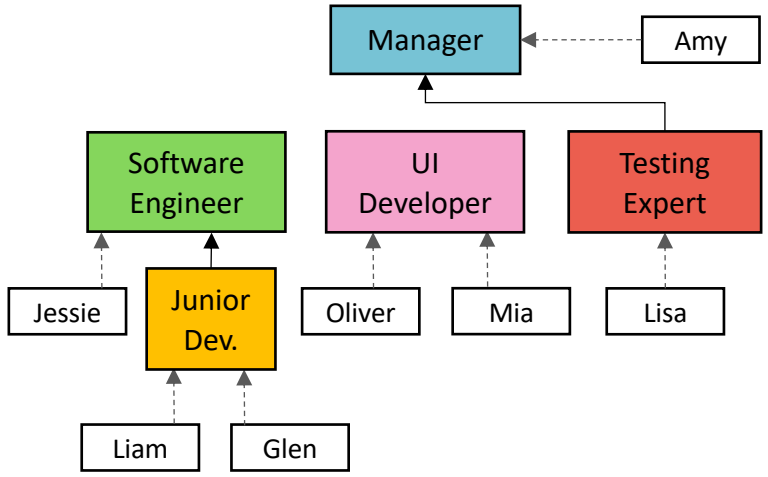


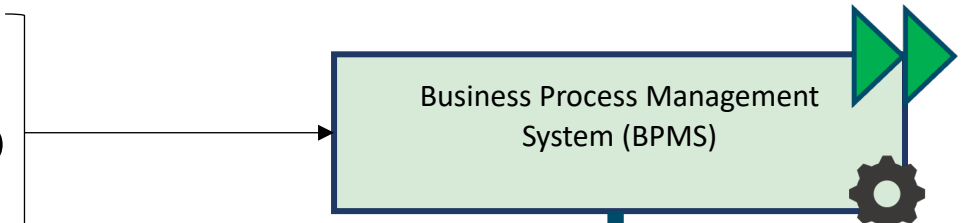
event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020
e6	1	start	Verify_backend	Lisa	07/2020
e7	1	end	Develop_frontend	Oliver	10/2020
e8	1	end	Verify_backend	Lisa	10/2020
e9	1	start	Verify_frontend	Lisa	10/2020
e10	1	end	Verify_frontend	Lisa	11/2020
e11	1	start	Deploy_system	Jessie	11/2020
e12	1	end	Deploy_system	Jessie	01/2021

- **Default activity durations**

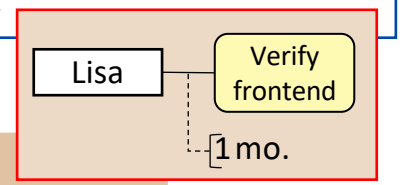
Running Business Processes
Mining a Temporal Model from the Event Log





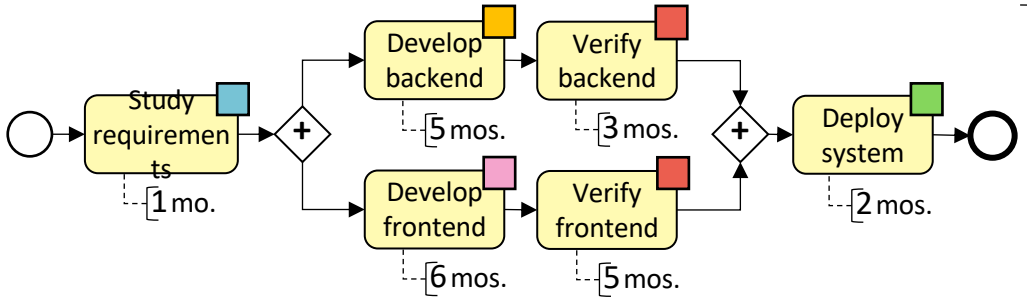


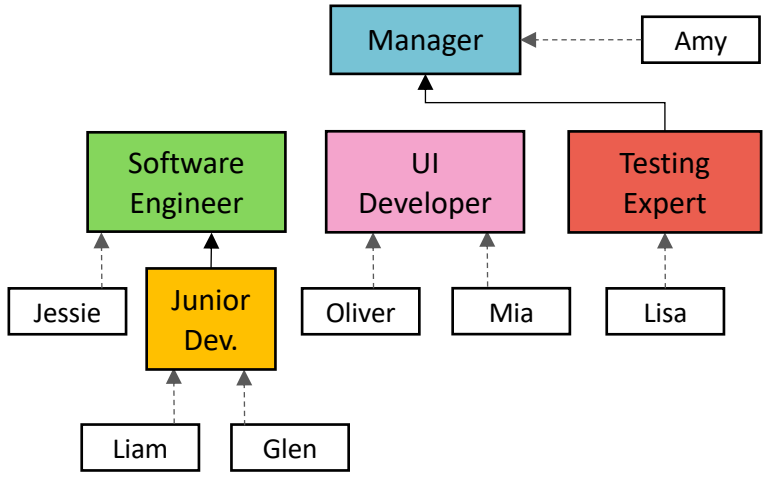
event_id	trace_id	event_type	activity	resource	time_stamp
e1	1	start	Study_requirements	Amy	01/2020
e2	1	end	Study_requirements	Amy	02/2020
e3	1	start	Develop_backend	Glen	02/2020
e4	1	start	Develop_frontend	Oliver	04/2020
e5	1	end	Develop_backend	Glen	07/2020
e6	1	start	Verify_backend	Lisa	07/2020
e7	1	end	Develop_frontend	Oliver	10/2020
e8	1	end	Verify_backend	Lisa	10/2020
e9	1	start	Verify_frontend	Lisa	10/2020
e10	1	end	Verify_frontend	Lisa	11/2020
e11	1	start	Deploy_system	Jessie	11/2020
e12	1	end	Deploy_system	Jessie	01/2021

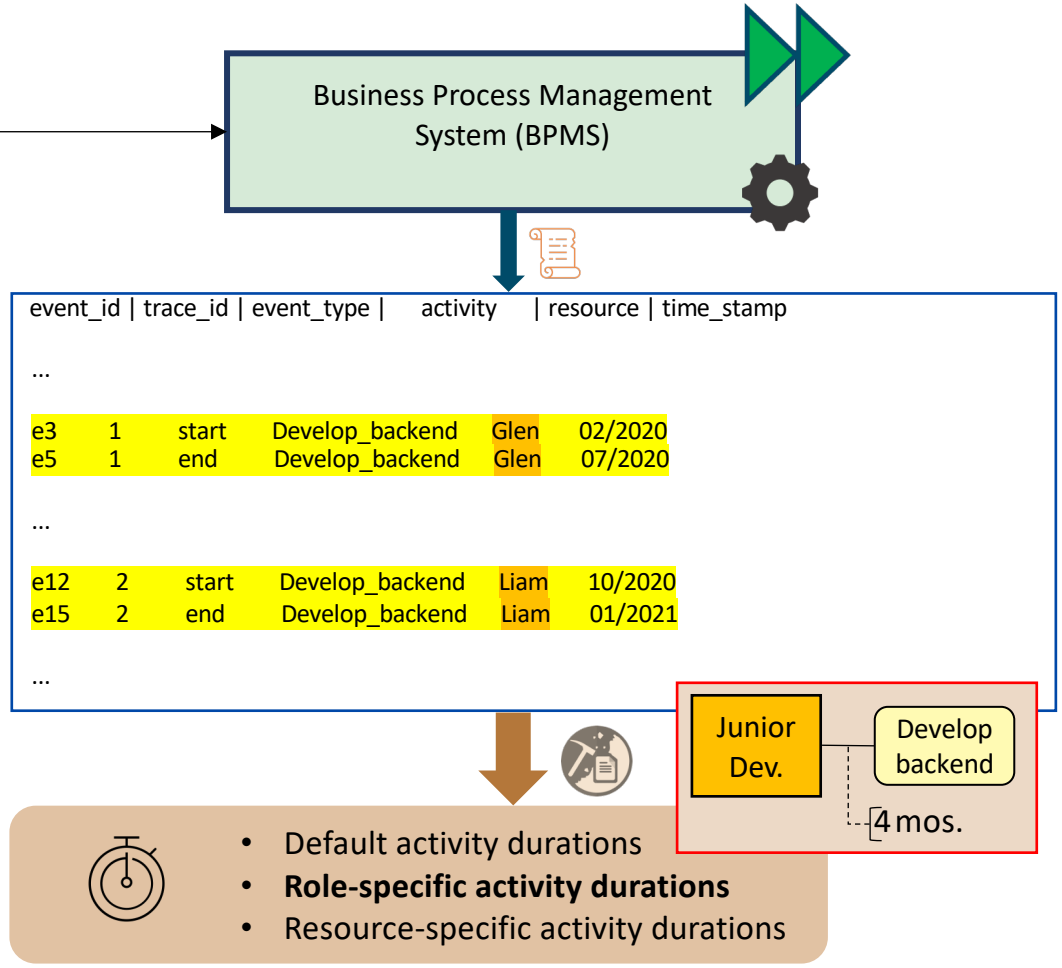


- Default activity durations
- **Resource-specific activity durations**

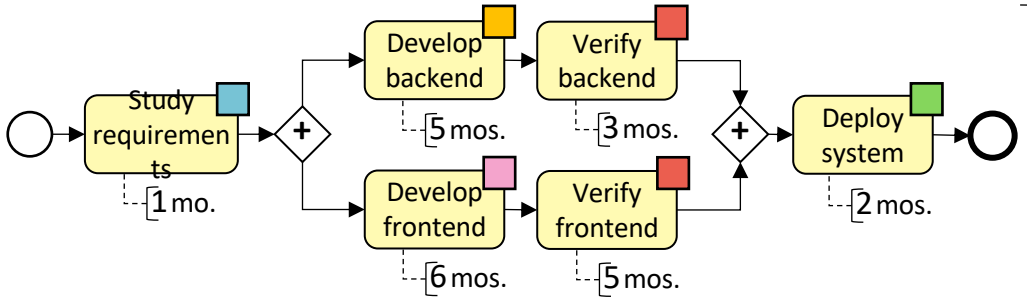
Running Business Processes
Mining a Temporal Model from the Event Log

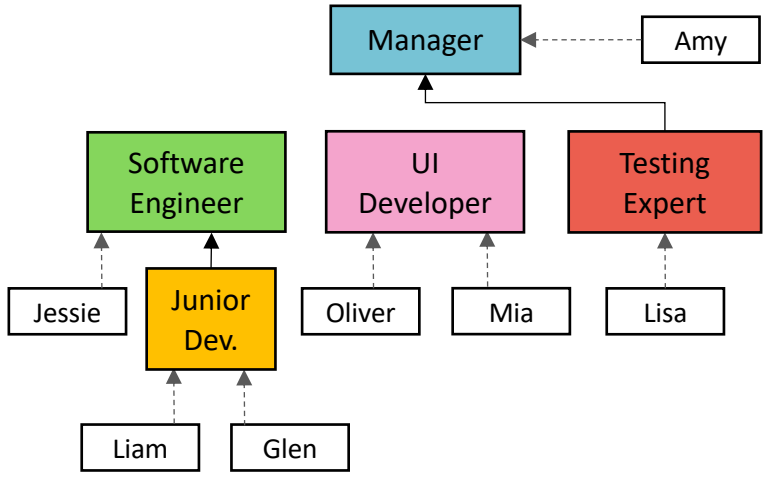


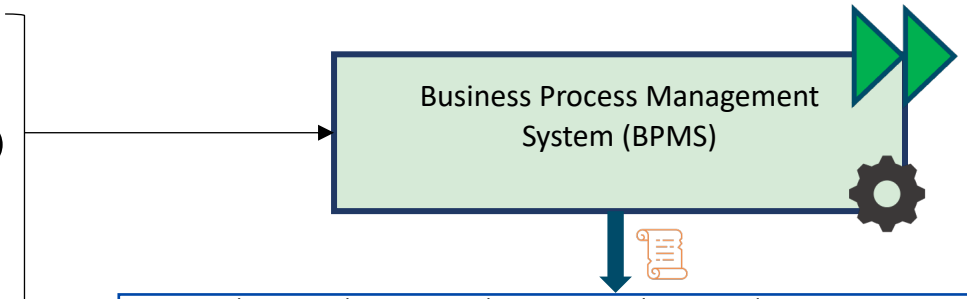




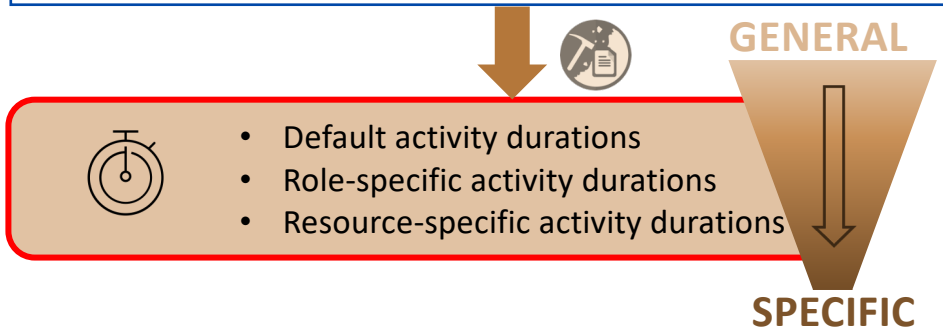
Running Business Processes
Mining a Temporal Model from the Event Log





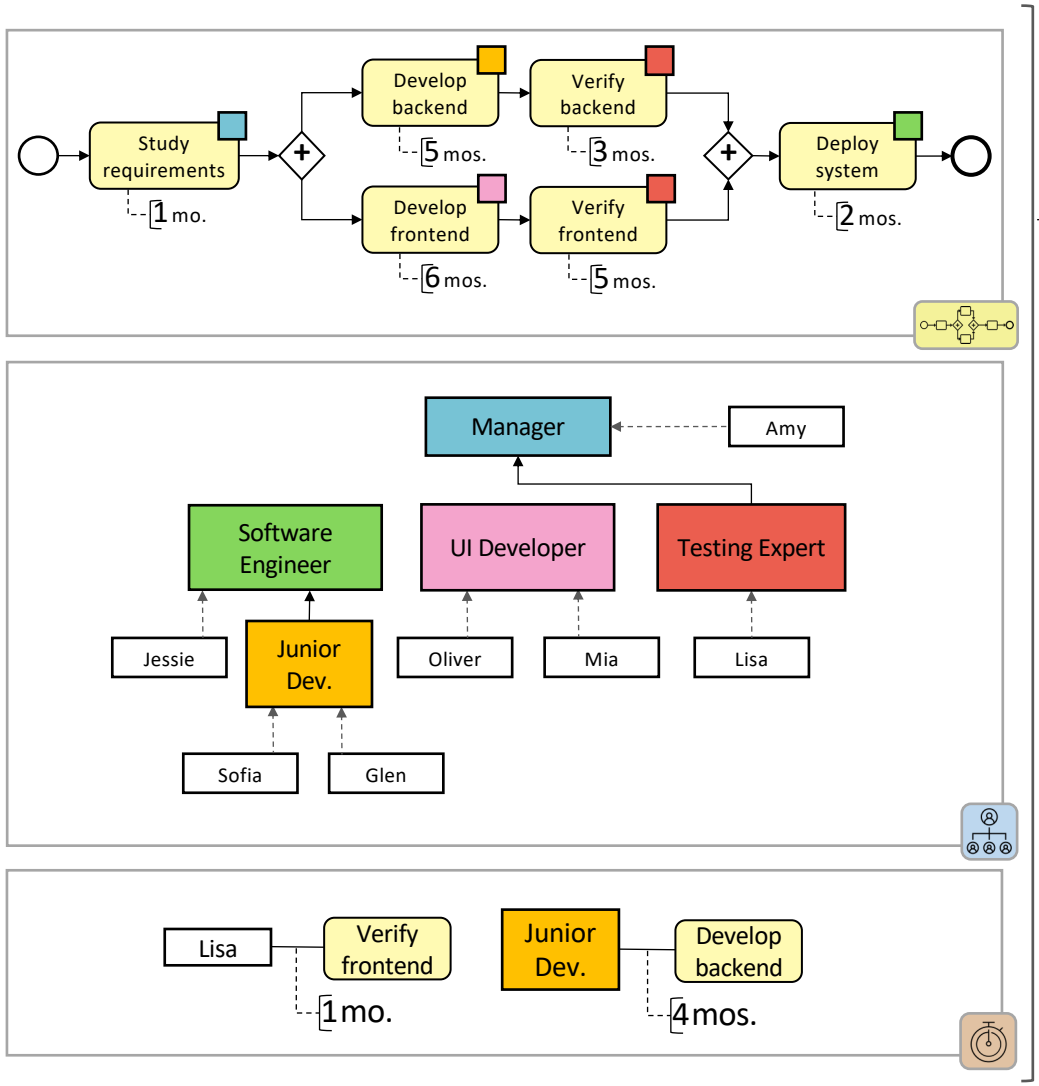


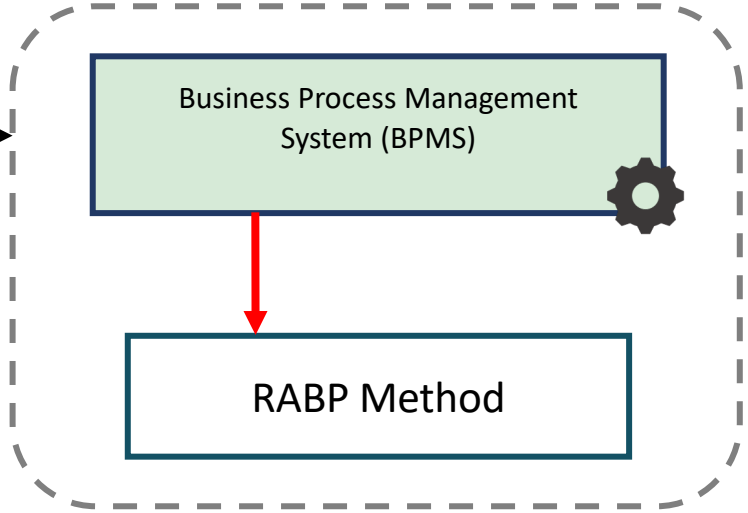
event_id	trace_id	event_type	activity	resource	time_stamp
...					
e3	1	start	Develop_backend	Glen	02/2020
e5	1	end	Develop_backend	Glen	07/2020
...					
e12	2	start	Develop_backend	Liam	10/2020
e15	2	end	Develop_backend	Liam	01/2021
...					

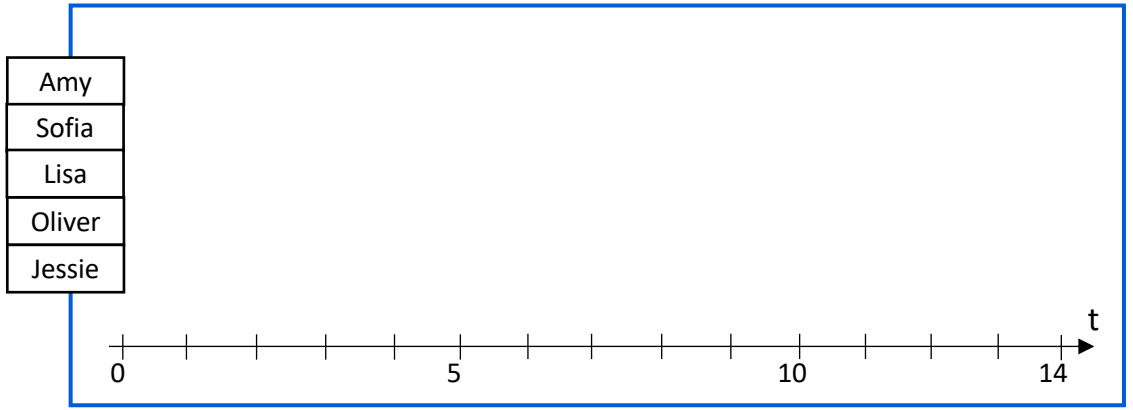


- Default activity durations
- Role-specific activity durations
- Resource-specific activity durations

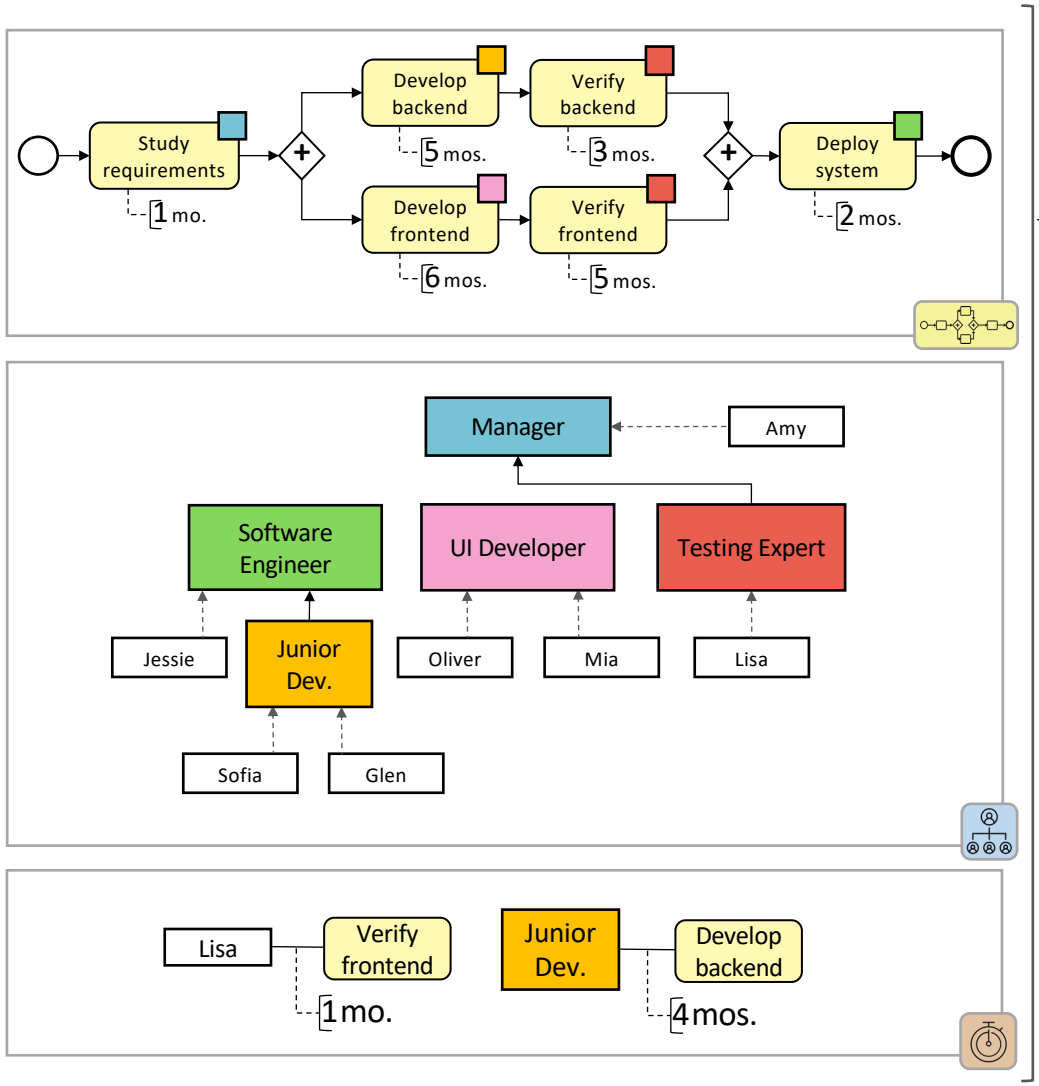
BPMS with RABP Support

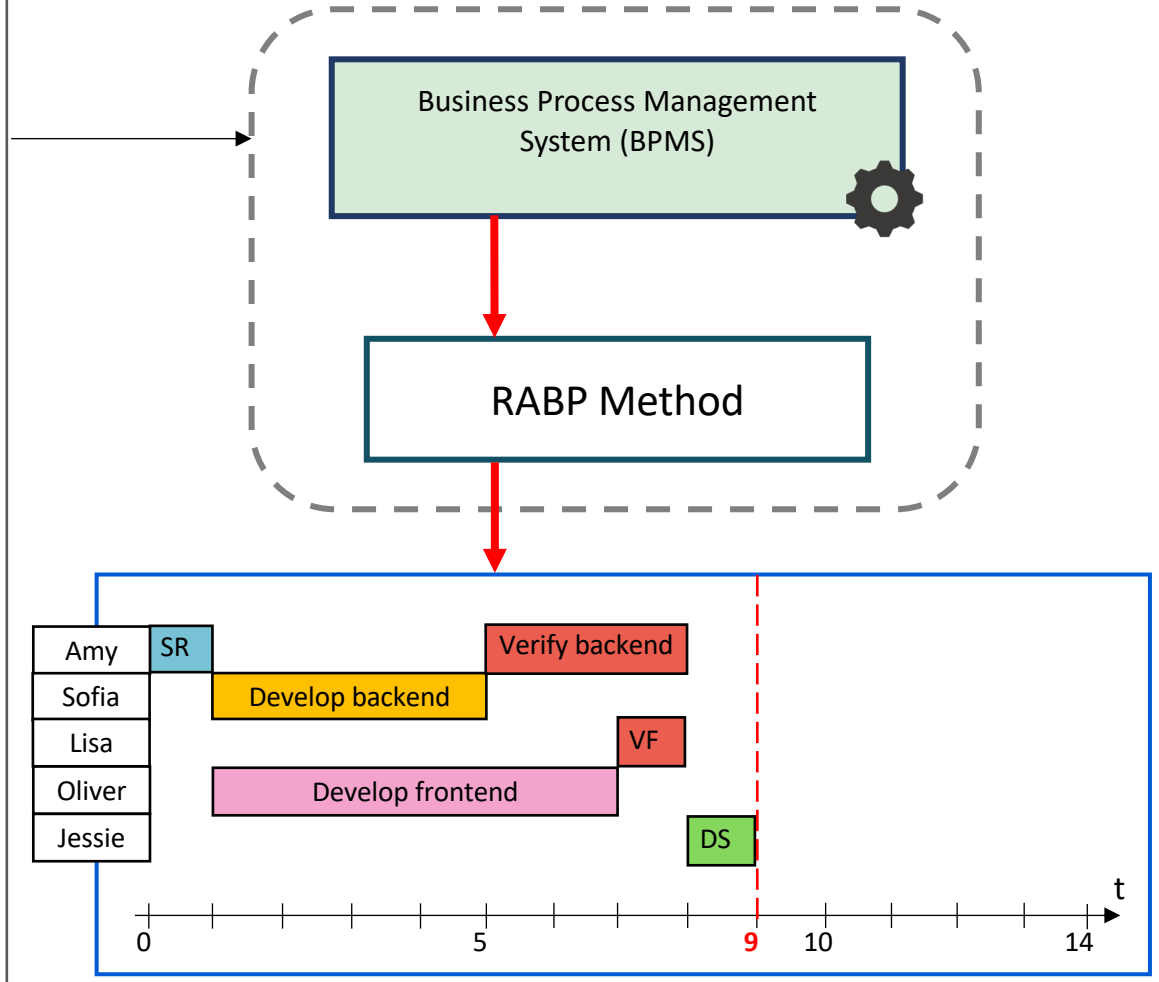




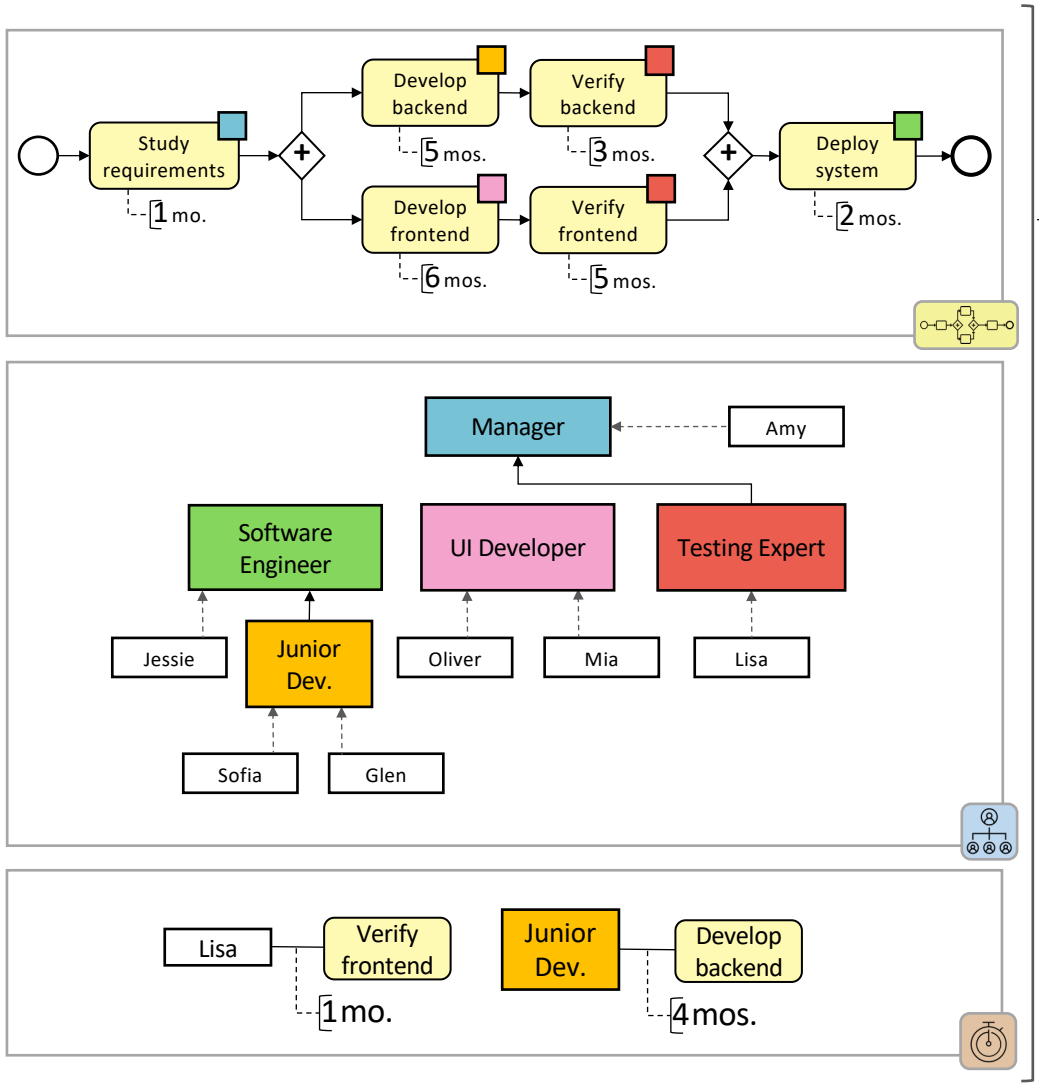


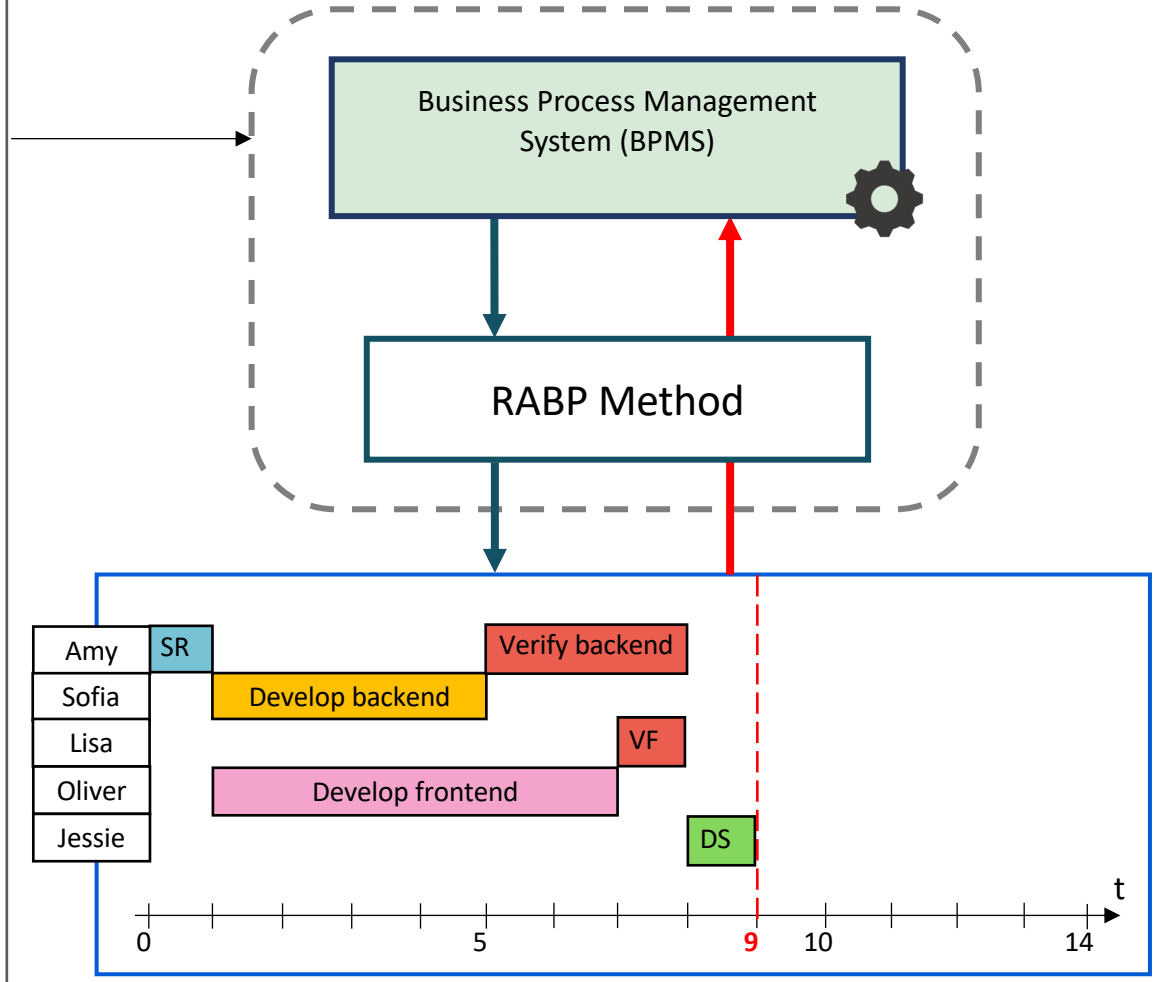
BPMS with RABP Support



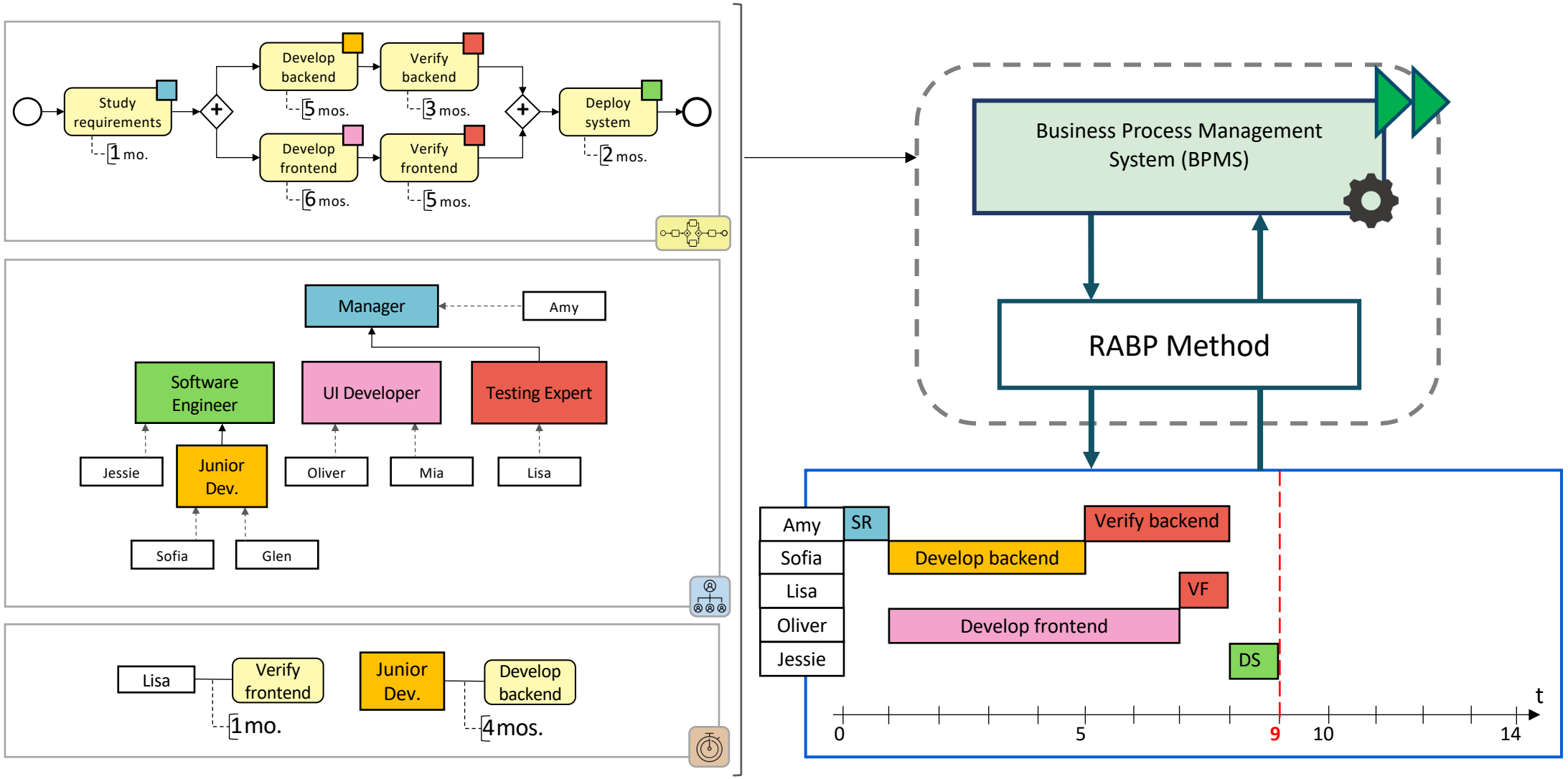


BPMS with RABP Support

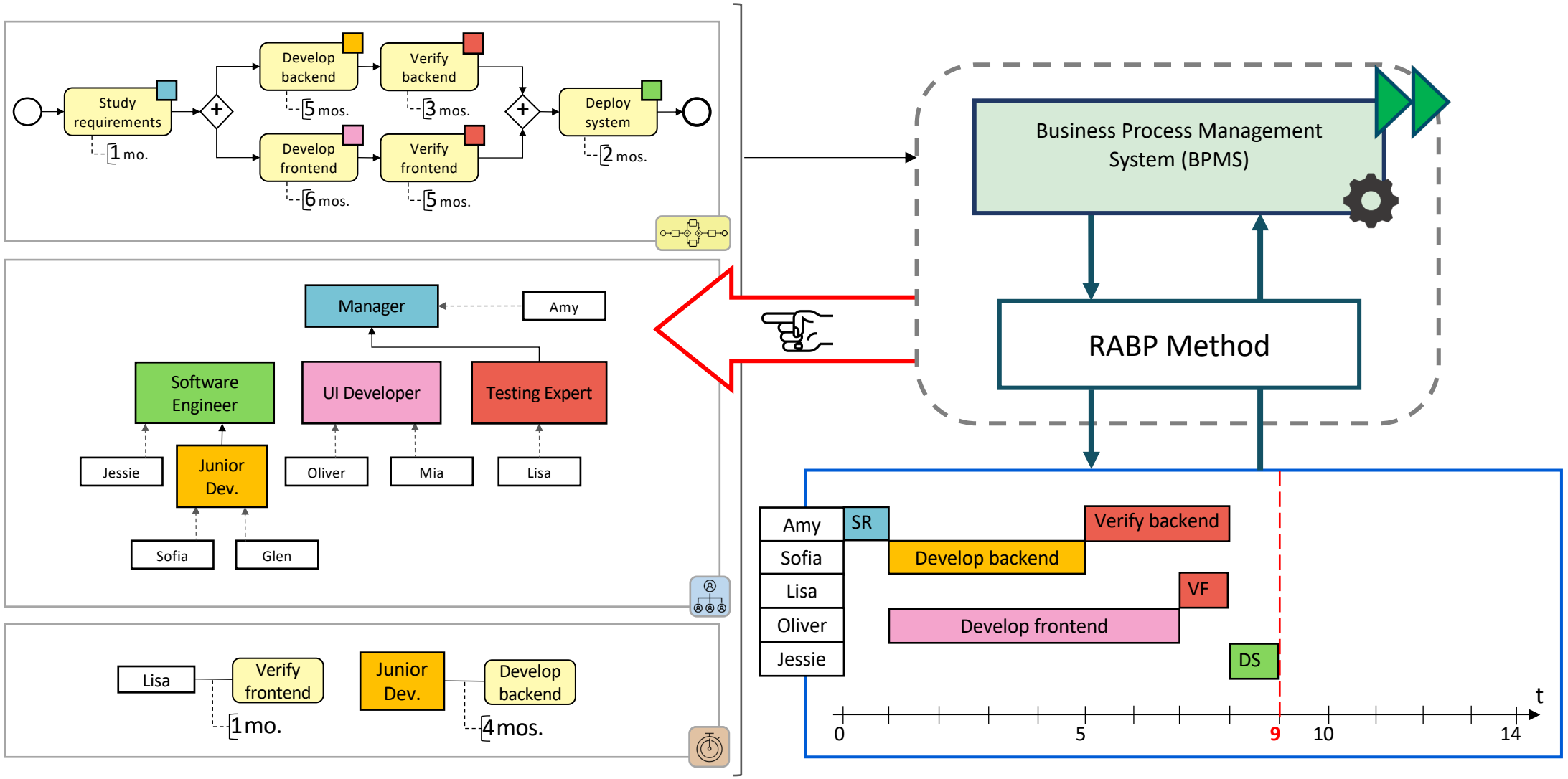




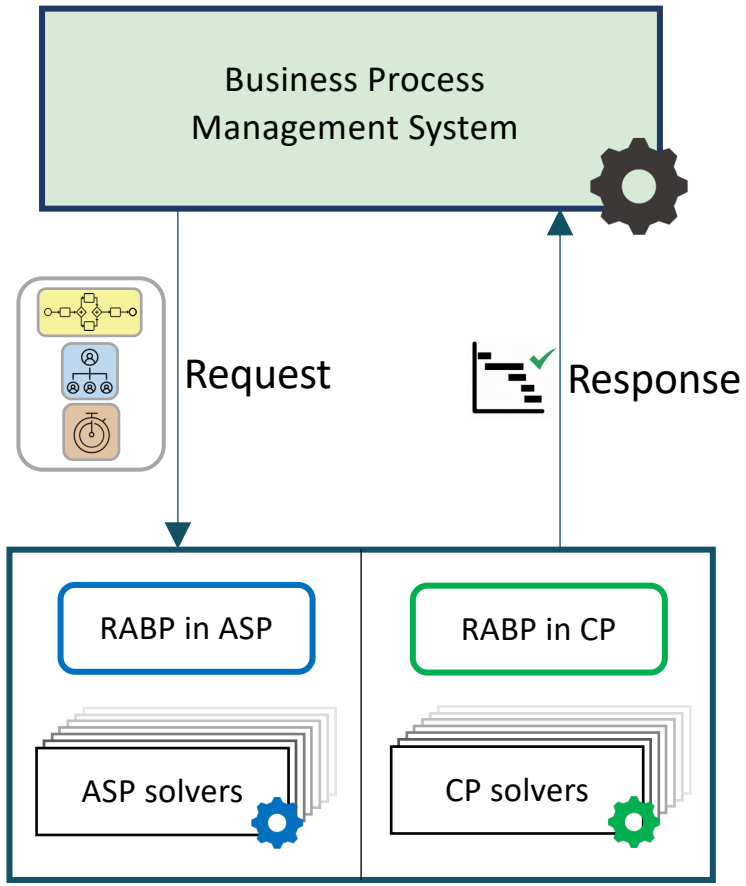
BPMS with RABP Support



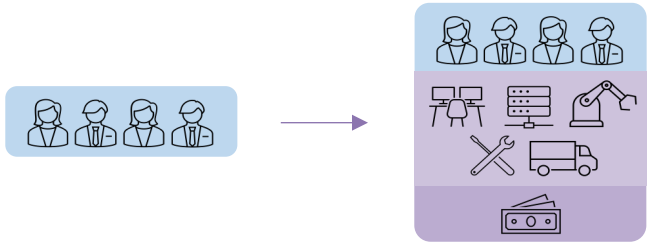
BPMS with RABP Support



Goals:



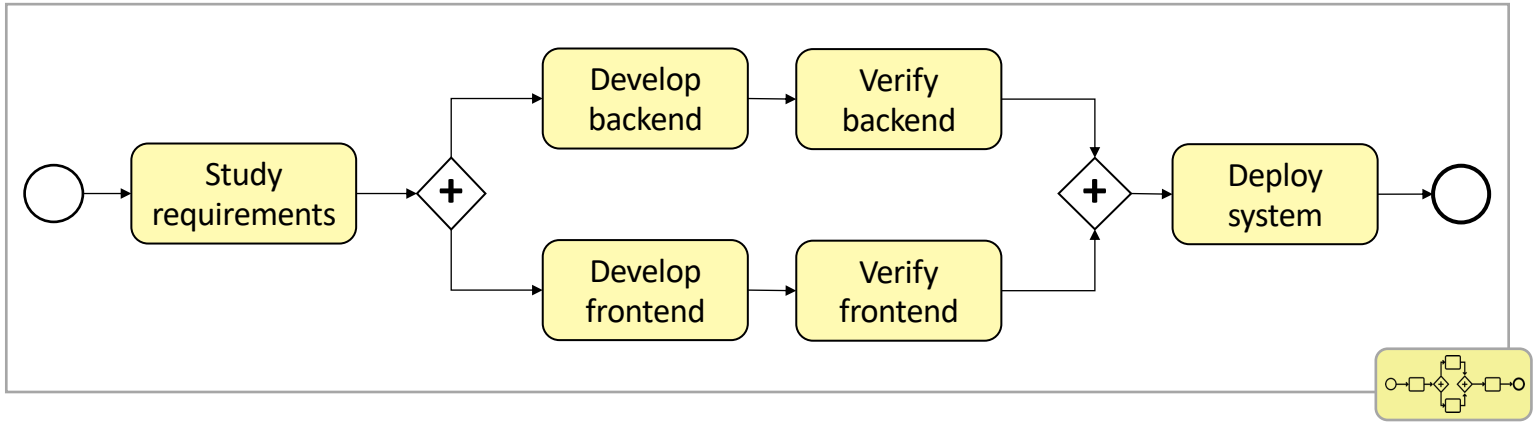
1. Representing a wide variety of resources in RABP



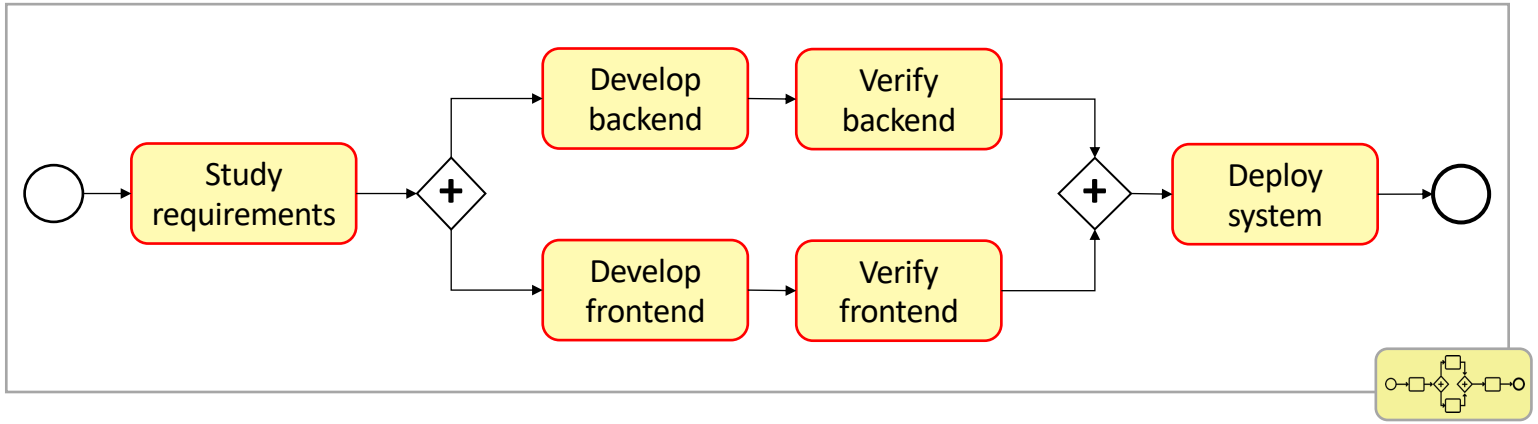
2. Selecting suitable KRR formalisms for implementing RABP
    - Declarative formalisms
        - Answer Set Programming (ASP)
        - Constraint Programming (CP)

3. Devising a realistic benchmark for testing RABP methods

BPM…. Ease of encoding in ASP

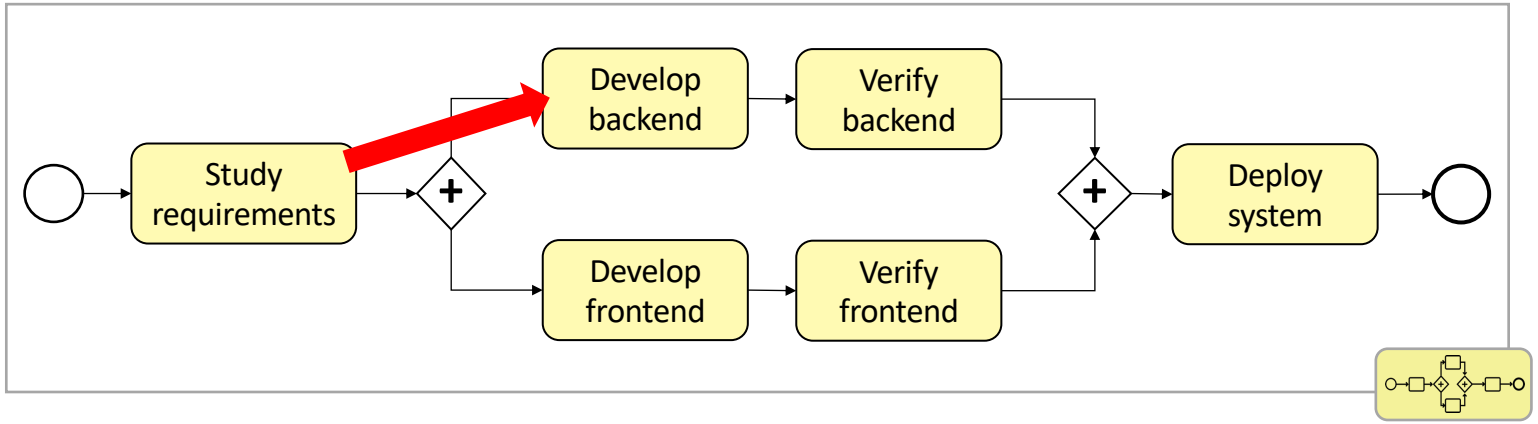


BPM…. Ease of encoding in ASP



activity(“StudyRequirements”).
activity(“DevelopBackend”).
activity(“VerifyBackend”).
activity(“DevelopFrontend”).
activity(“VerifyFrontend”).
activity(“DeploySystem”).

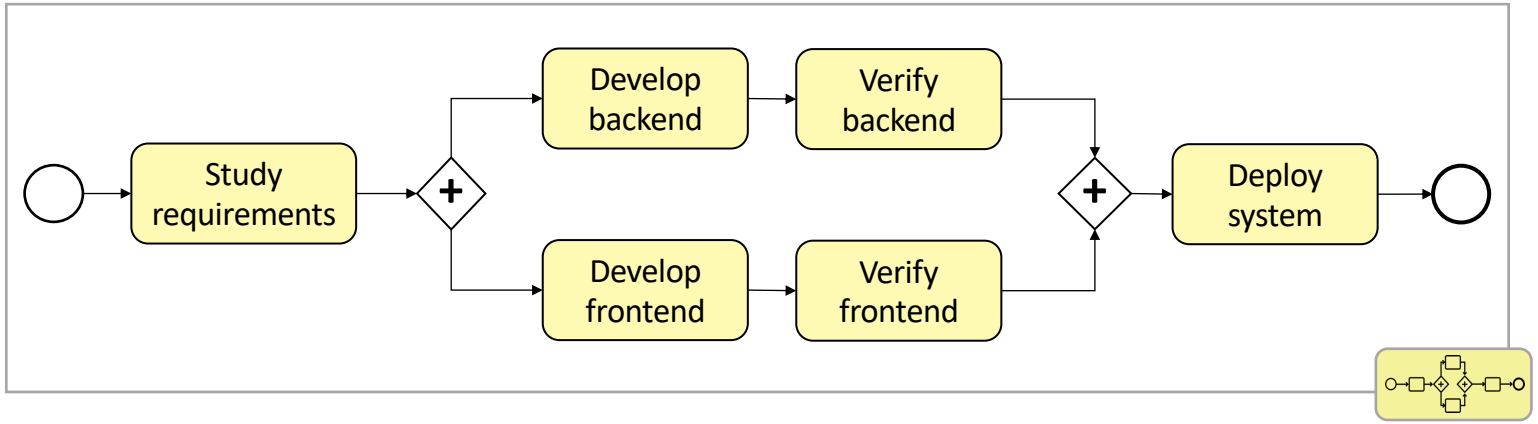
BPM…. Ease of encoding in ASP



activity(“StudyRequirements”).
activity(“DevelopBackend”).
activity(“VerifyBackend”).
activity(“DevelopFrontend”).
activity(“VerifyFrontend”).
activity(“DeploySystem”).

dprec(“StudyRequirements”, “DevelopBackend”).

BPM…. Ease of encoding in ASP

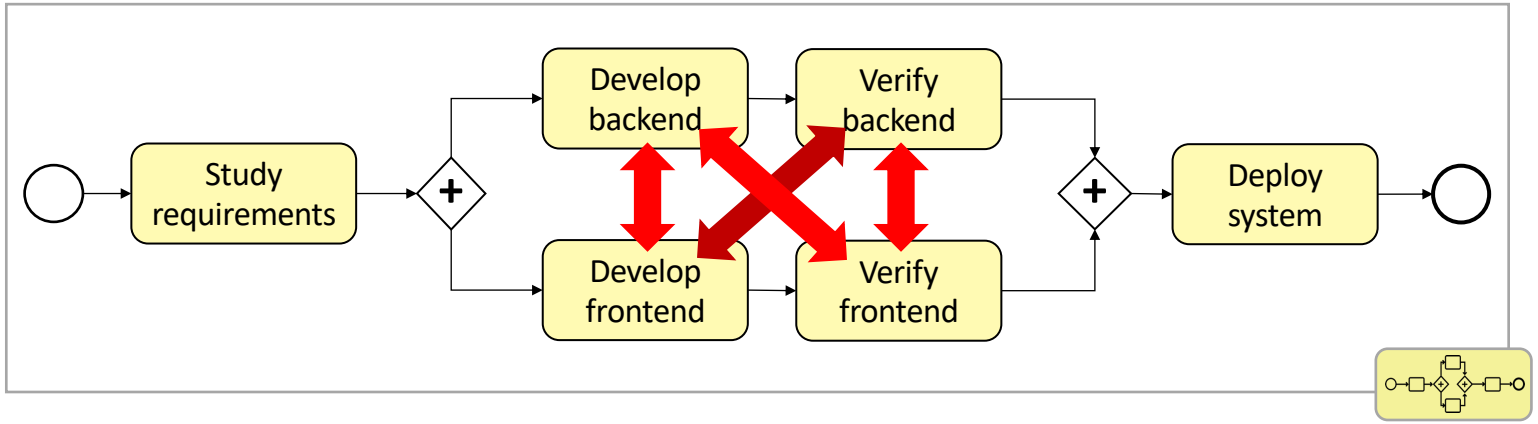



```
activity(“StudyRequirements”).
activity(“DevelopBackend”).
activity(“VerifyBackend”).
activity(“DevelopFrontend”).
activity(“VerifyFrontend”).
activity(“DeploySystem”).
```

```
dprec(“StudyRequirements”, “DevelopBackend”).
dprec(“StudyRequirements”, “DevelopFrontend”).
dprec(“DevelopBackend”, “VerifyBackend”).
dprec(“DevelopFrontend”, “VerifyFrontend”).
dprec(“VerifyBackend”, “DeploySystem”).
dprec(“VerifyFrontend”, “DeploySystem”).
```

Contributions to RQ 1

Representation of Processes in ASP

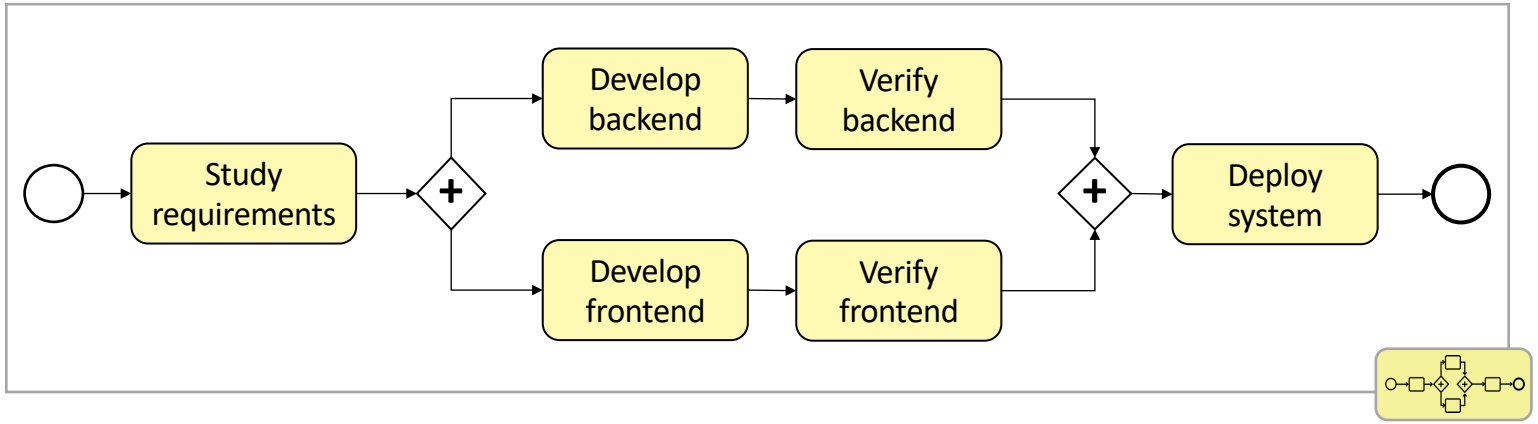



```
activity(“StudyRequirements”).
activity(“DevelopBackend”).
activity(“VerifyBackend”).
activity(“DevelopFrontend”).
activity(“VerifyFrontend”).
activity(“DeploySystem”).
```

```
dprec(“StudyRequirements”, “DevelopBackend”).
dprec(“StudyRequirements”, “DevelopFrontend”).
dprec(“DevelopBackend”, “VerifyBackend”).
dprec(“DevelopFrontend”, “VerifyFrontend”).
dprec(“VerifyBackend”, “DeploySystem”).
dprec(“VerifyFrontend”, “DeploySystem”).
```

```
conc(“DevelopBackend”, “DevelopFrontend”).
conc(“DevelopBackend”, “VerifyFrontend”).
conc(“DevelopFrontend”, “DevelopBackend”).
conc(“DevelopFrontend”, “VerifyBackend”).
```

BPM…. Ease of encoding in ASP

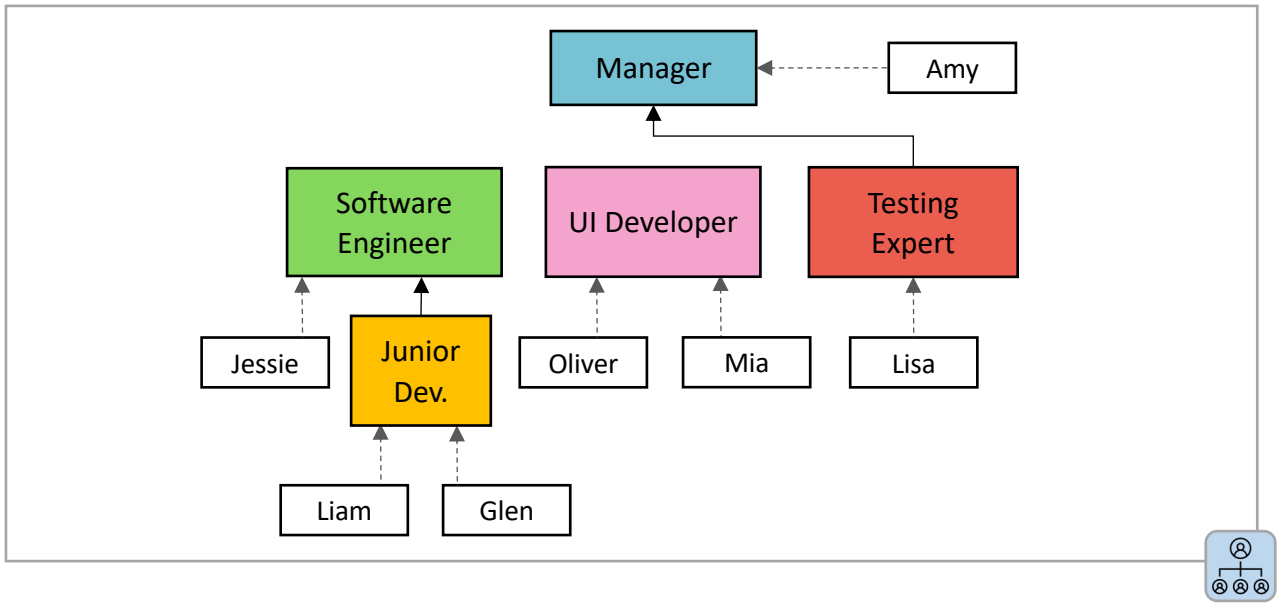



```
activity(“StudyRequirements”).
activity(“DevelopBackend”).
activity(“VerifyBackend”).
activity(“DevelopFrontend”).
activity(“VerifyFrontend”).
activity(“DeploySystem”).
```

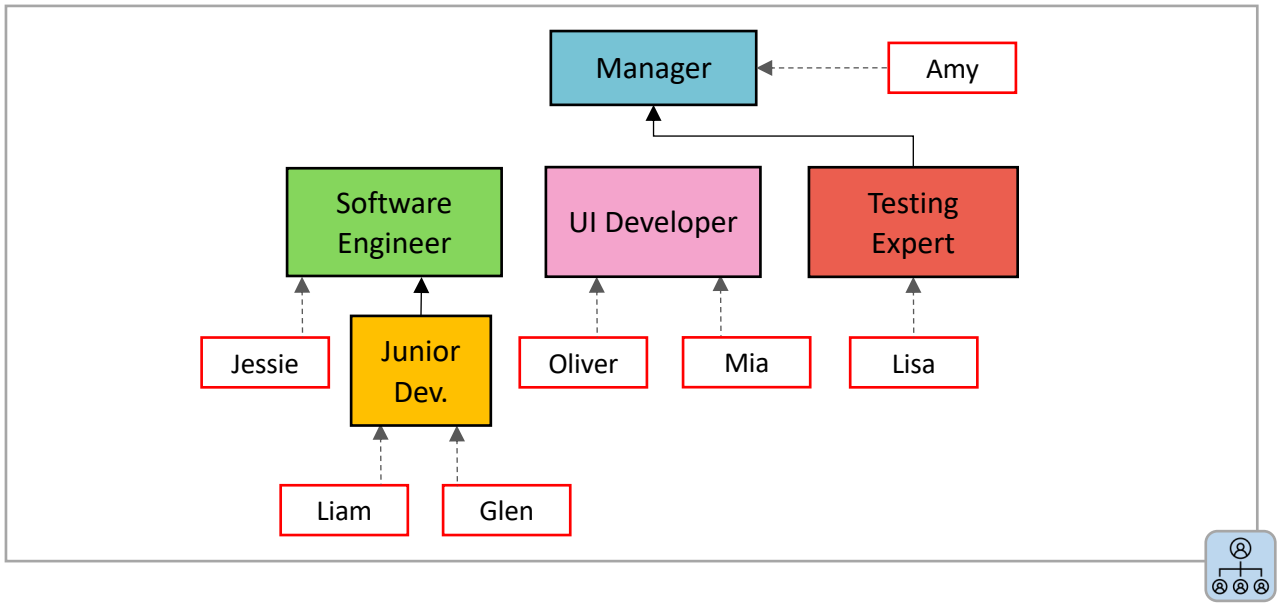
```
dprec(“StudyRequirements”, “DevelopBackend”).
dprec(“StudyRequirements”, “DevelopFrontend”).
dprec(“DevelopBackend”, “VerifyBackend”).
dprec(“DevelopFrontend”, “VerifyFrontend”).
dprec(“VerifyBackend”, “DeploySystem”).
dprec(“VerifyFrontend”, “DeploySystem”).
```

```
conc(“DevelopBackend”, “DevelopFrontend”).
conc(“DevelopBackend”, “VerifyFrontend”).
conc(“DevelopFrontend”, “DevelopBackend”).
conc(“DevelopFrontend”, “VerifyBackend”).
```

BPM…. Ease of encoding in ASP

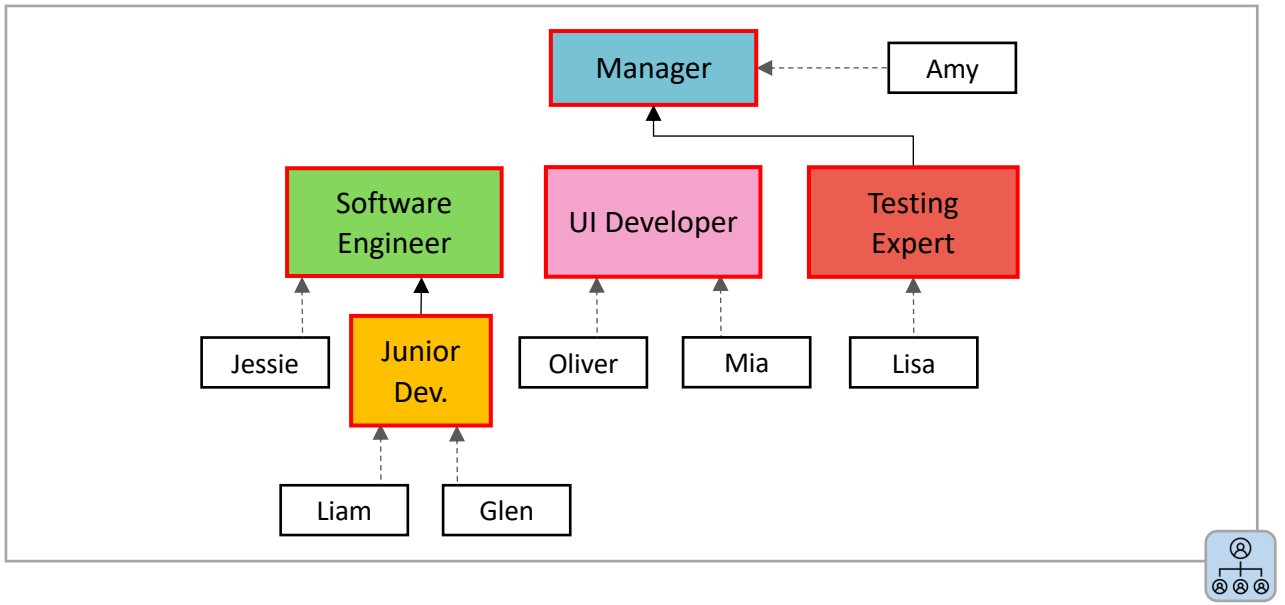


BPM…. Ease of encoding in ASP



resource(“Amy”).
resource(“Jessie”).
resource(“Liam”).
resource(“Glen”).
resource(“Oliver”).
resource(“Mia”).
resource(“Lisa”).

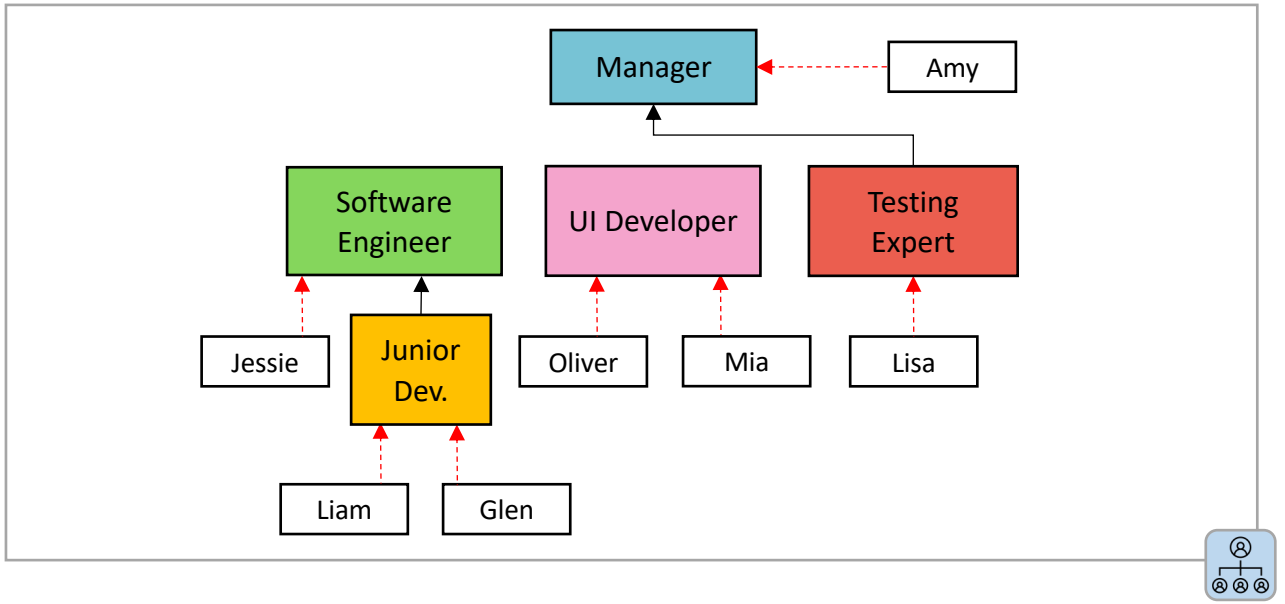
BPM…. Ease of encoding in ASP



resource(“Amy”).
resource(“Jessie”).
resource(“Liam”).
resource(“Glen”).
resource(“Oliver”).
resource(“Mia”).
resource(“Lisa”).

role(“Manager”).
role(“SoftwareEngineer”).
role(“JuniorDeveloper”).
role(“UIDeveloper”).
role(“TestingExpert”).

BPM…. Ease of encoding in ASP

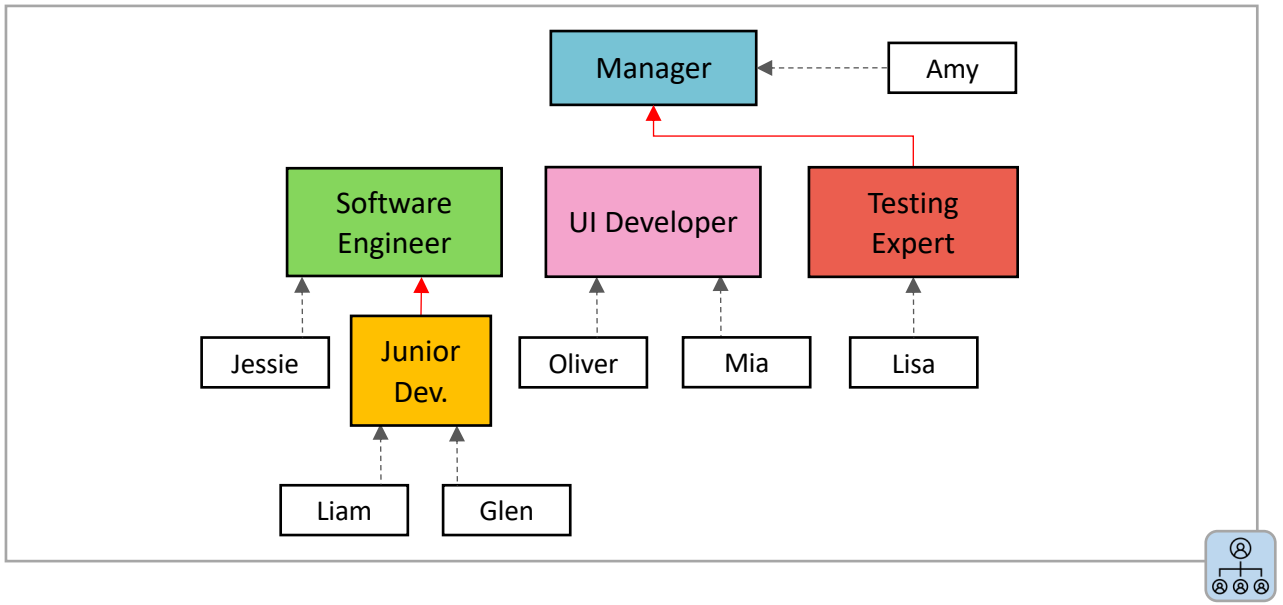



```
resource(“Amy”).
resource(“Jessie”).
resource(“Liam”).
resource(“Glen”).
resource(“Oliver”).
resource(“Mia”).
resource(“Lisa”).
```

```
role(“Manager”).
role(“SoftwareEngineer”).
role(“JuniorDeveloper”).
role(“UIDeveloper”).
role(“TestingExpert”).
```

```
rlAC(“Amy”, “Manager”).
rlAC(“Jessie”,”SoftwareEngineer”).
rlAC(“Liam”,”JuniorDeveloper”).
rlAC(“Glen”,”JuniorDeveloper”).
rlAC(“Oliver”,”UIDeveloper”).
rlAC(“Mia”,”UIDeveloper”).
rlAC(“Lisa”,”TestingExpert”).
```

BPM…. Ease of encoding in ASP



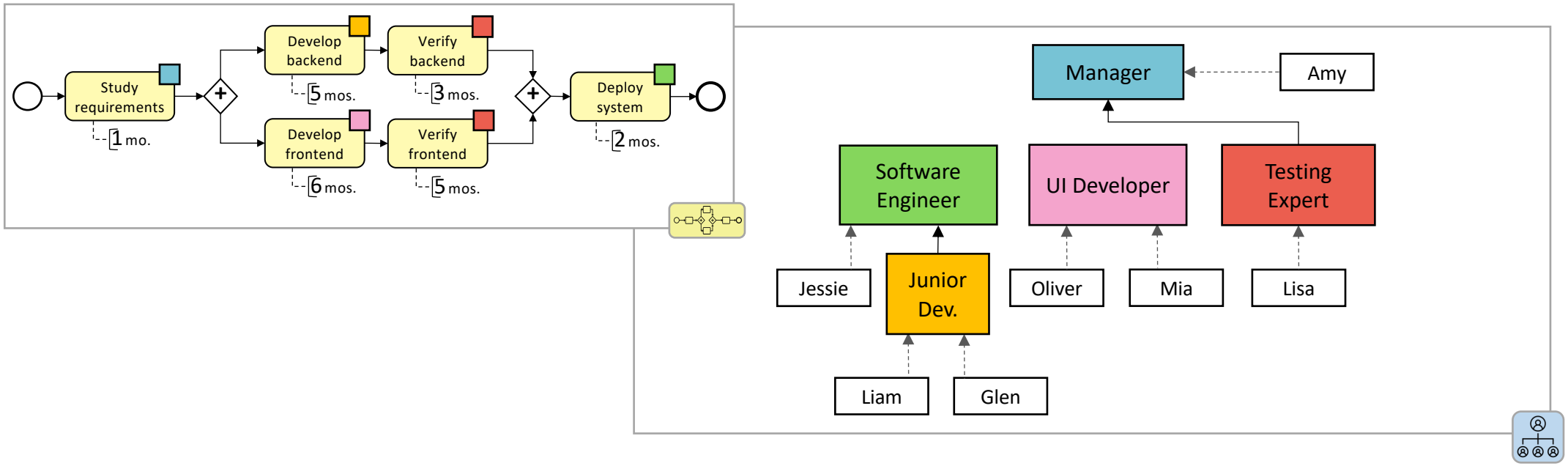
resource(“Amy”).
resource(“Jessie”).
resource(“Liam”).
resource(“Glen”).
resource(“Oliver”).
resource(“Mia”).
resource(“Lisa”).

role(“Manager”).
role(“SoftwareEngineer”).
role(“JuniorDeveloper”).
role(“UIDeveloper”).
role(“TestingExpert”).

rIAC(“Amy”, “Manager”).
rIAC(“Jessie”,”SoftwareEngineer”).
rIAC(“Liam”,”JuniorDeveloper”).
rIAC(“Glen”,”JuniorDeveloper”).
rIAC(“Oliver”,”UIDeveloper”).
rIAC(“Mia”,”UIDeveloper”).
rIAC(“Lisa”,”TestingExpert”).

lIAC(“Manager”, “TestingExpert”).
lIAC(”SoftwareEngineer”,“JuniorDeveloper”).

BPM…. Ease of encoding in ASP



resource(“Amy”).
resource(“Jessie”).
resource(“Liam”).
resource(“Glen”).
resource(“Oliver”).
resource(“Mia”).
resource(“Lisa”).

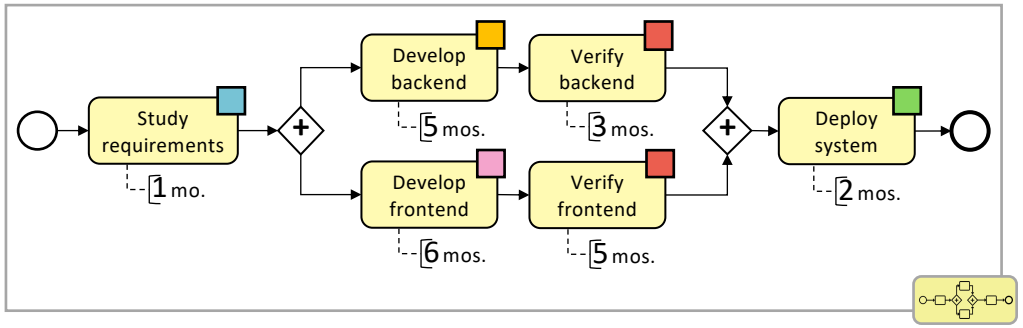
role(“Manager”).
role(“SoftwareEngineer”).
role(“JuniorDeveloper”).
role(“UIDeveloper”).
role(“TestingExpert”).

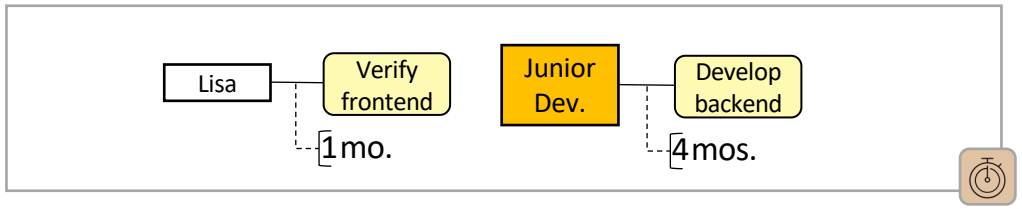
rlAC(“Amy”, “Manager”).
rlAC(“Jessie”,”SoftwareEngineer”).
rlAC(“Liam”,”JuniorDeveloper”).
rlAC(“Glen”,”JuniorDeveloper”).
rlAC(“Oliver”,”UIDeveloper”).
rlAC(“Mia”,”UIDeveloper”).
rlAC(“Lisa”,”TestingExpert”).

llAC(“Manager”, “TestingExpert”).
llAC(”SoftwareEngineer”,“JuniorDeveloper”).

alAC(“StudyRequirements”, “Manager”).
alAC(“DevelopBackend”, “JuniorDeveloper”).
alAC(“VerifyBackend”, “TestingExpert”).
alAC(“DevelopFrontend”, “UIDeveloper”).
alAC(“VerifyFrontend”, “TestingExpert”).
alAC(“DeploySystem”, “SoftwareEngineer”).

BPM…. Ease of encoding in ASP






```
defaultDuration(“StudyRequirements”,1).
defaultDuration(“DevelopBackend”,5).
defaultDuration(“VerifyBackend”,3).
defaultDuration(“DevelopFrontend”,6).
defaultDuration(“VerifyFrontend”,5).
defaultDuration(“DeploySystem”,2).
```

```
rsaDuration(“Lisa”,”VerifyFrontend”,1).
lsaDuration(“JuniorDeveloper”,”DevelopBackend”,4).
```

Specific

- Resource-specific activity durations
- Role-specific activity durations
- Default activity durations

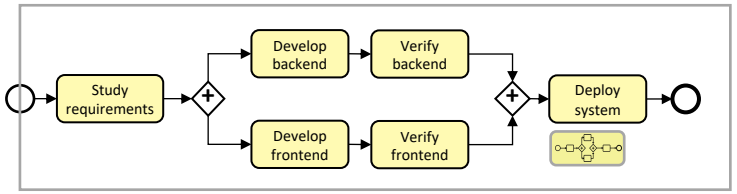
General

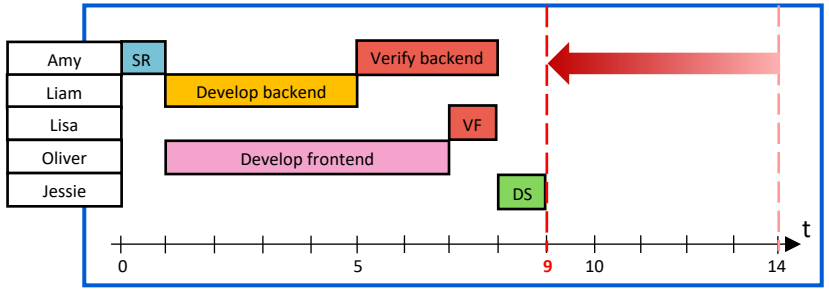
- Resource-to-role assignments
- Activity-to-role assignments
- Role-to-role assignments

```
allowedRAD(R,A,D) :- rsaDuration(R,A,D), rlAC(R,L), alAC(A,L).
allowedRAD(R,A,D) :- lsaDuration(L,A,D), not rsaDuration(R,A,_), rlAC(R,L), alAC(A,L).
allowedRAD(R,A,D) :- defaultDuration(A,D), not rsaDuration(R,A,_), not lsaDuration(L,A,_), rlAC(R,L), alAC(A,L).
```

BPM.... Guess and Check

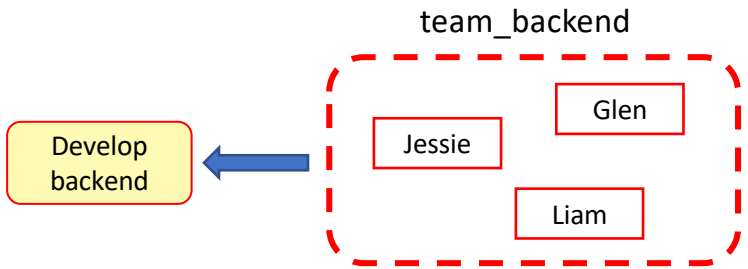
```
% generate allocations
1<={allocation(R,A,S,C): time(S), time(C), allowedRAD(R,A,D), C=S+D}<=1 :- activity(A).

% check for scheduling of the preceding activities
:- dPrec(A1,A2), allocation(_,A1,_,C1), allocation(_,A2,S2,_), C1>S2.

% check for scheduling of the concurrent activities
:- conc(A1,A2), allocation(R,A1,S1,_), allocation(R,A2,S2,C2), S2<=S1, C2>S1, A1<A2.
:- conc(A1,A2), allocation(R,A1,_,C1), allocation(R,A2,S2,C2), S2<C1, C2>=C1, A1<A2.
:- conc(A1,A2), allocation(R,A1,S1,C1), allocation(R,A2,S2,C2), S2>S1, C2<C1, A1<A2.

% minimize makespan
:~ makespan(U). [U]
```

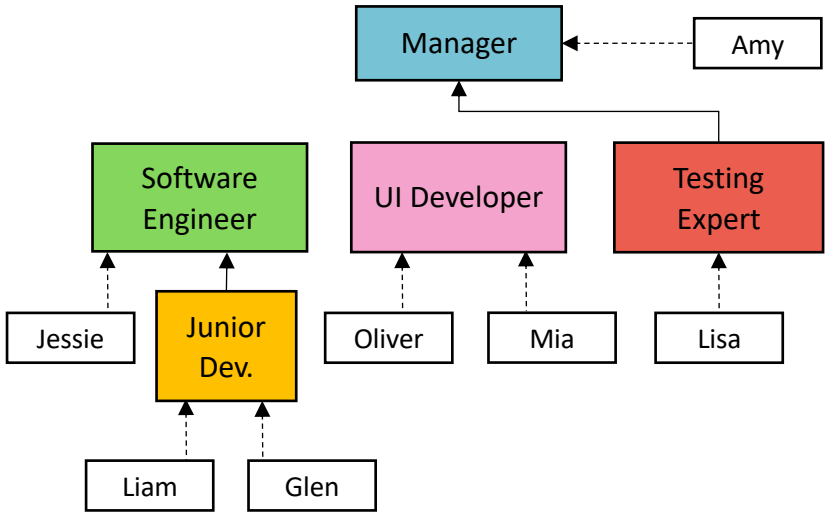




BPM … Easily Extensible encoding

- Teams
  - **E.g.,** team(“team_backend”,R) :- rlAC(R,”JuniorDeveloper”).
    team(“team_backend”,R) :- rlAC(R,”SoftwareEngineer”).
    tRequirement(“DevelopBackend”,”team_backend”).

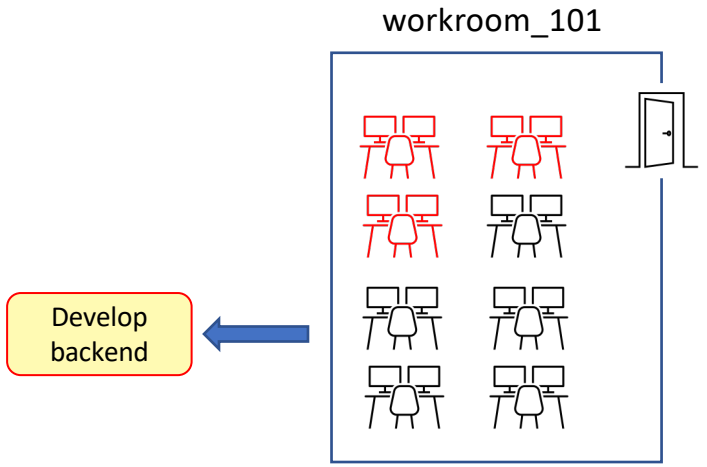




BPM … Easily Extensible encoding

- Teams
  - **E.g.,** team(“team_backend”,R) :- rlAC(R,”JuniorDeveloper”).
    team(“team_backend”,R) :- rlAC(R,”SoftwareEngineer”).
    tRequirement(“DevelopBackend”,”team_backend”).

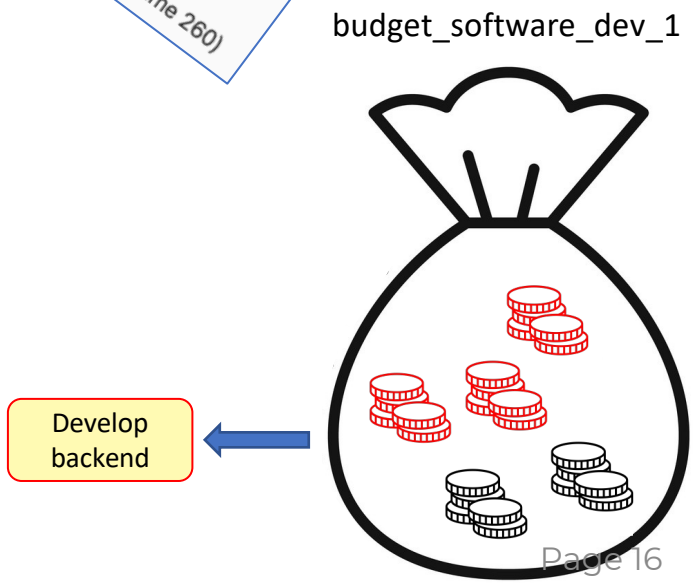
- Partially-renewable resources
  - **E.g.,** pResource(“workroom_101”,8).
    pRequirement(“DevelopBackend”,”workroom_101”, N) :-
    N = #count{R: team(“team_backend”,R)}.



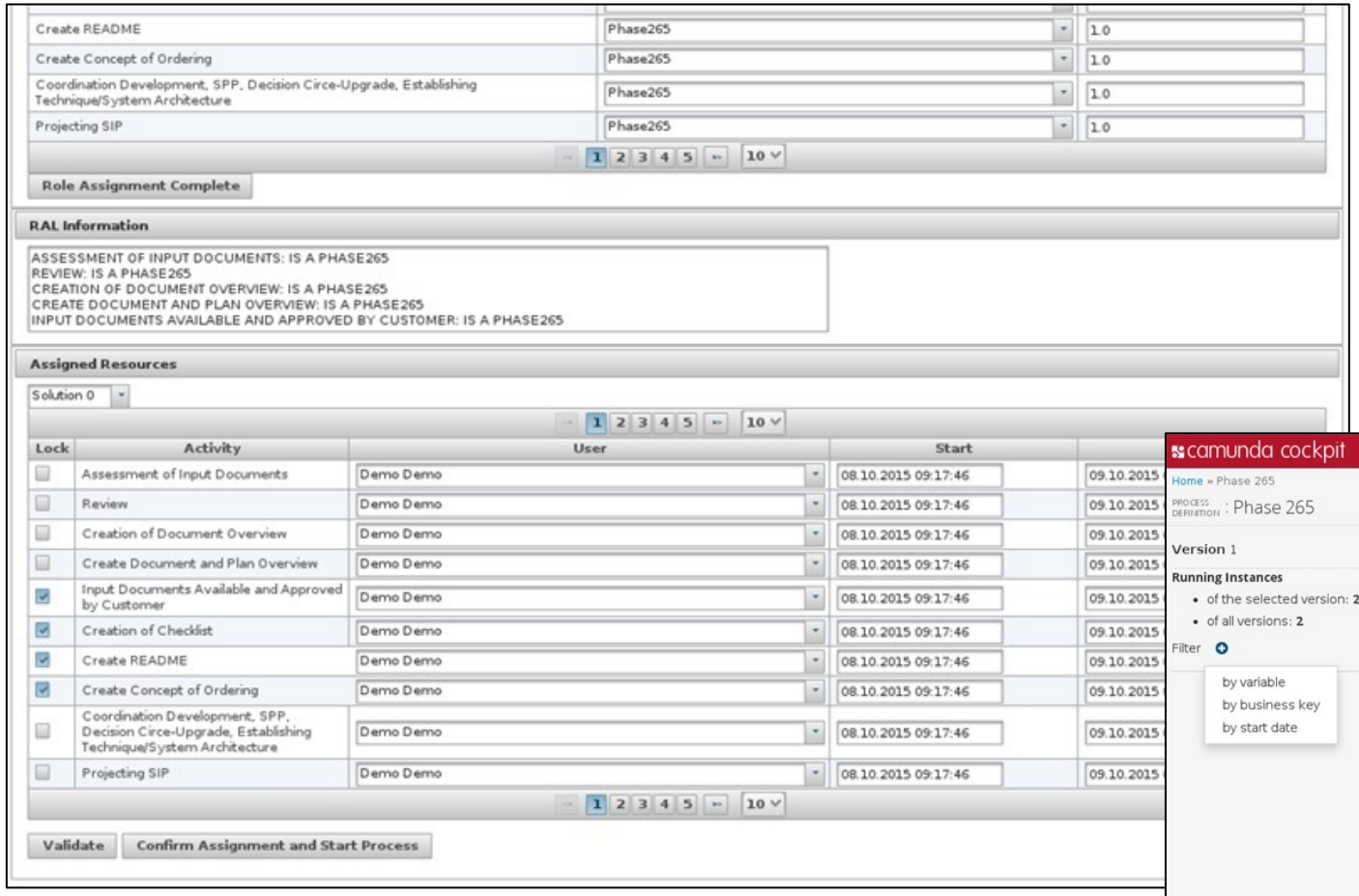
BPM … Easily Extensible encoding

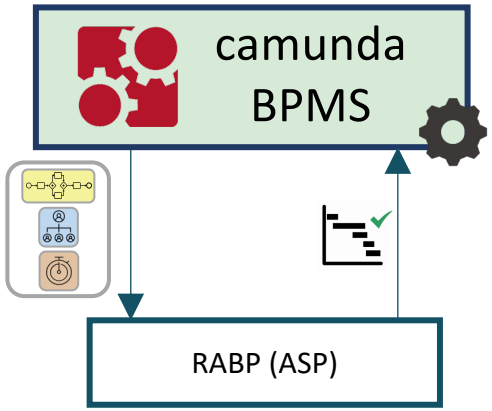
- Teams
  - **E.g.,** team(“team_backend”,R) :- rIAC(R,”JuniorDeveloper”).
    team(“team_backend”,R) :- rIAC(R,”SoftwareEngineer”).
    tRequirement(“DevelopBackend”,”team_backend”).

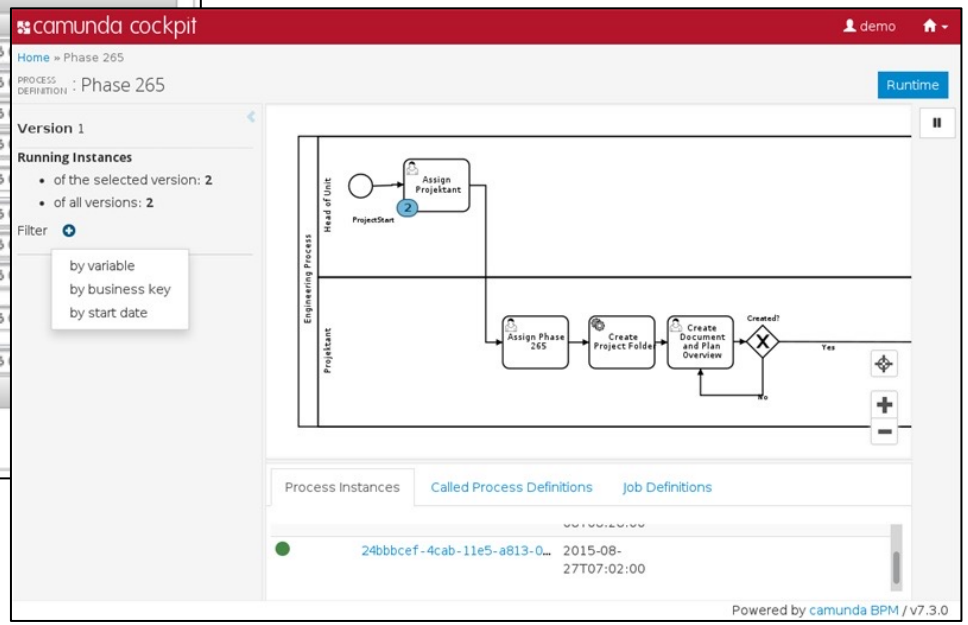
- Partially-renewable resources
  - **E.g.,** pResource(“workroom_101”,8).
    pRequirement(“DevelopBackend”,”workroom_101”, N) :-
    N = #count{R: team(“team_backend”,R)}.
- Non-renewable resources
  - **E.g.,** nResource(“budget_software_dev_1”,50000).
    nRequirement(“DevelopBackend”,”budget_software_dev_1”,N) :-
    TPM = #sum{PM,R: allocate(R,”DevelopBackend”,S,C), PM=C-S},
    costPersonMonth(CPM), N = TPM * CPM.



RABP in Practice: Camunda BPMS Integration



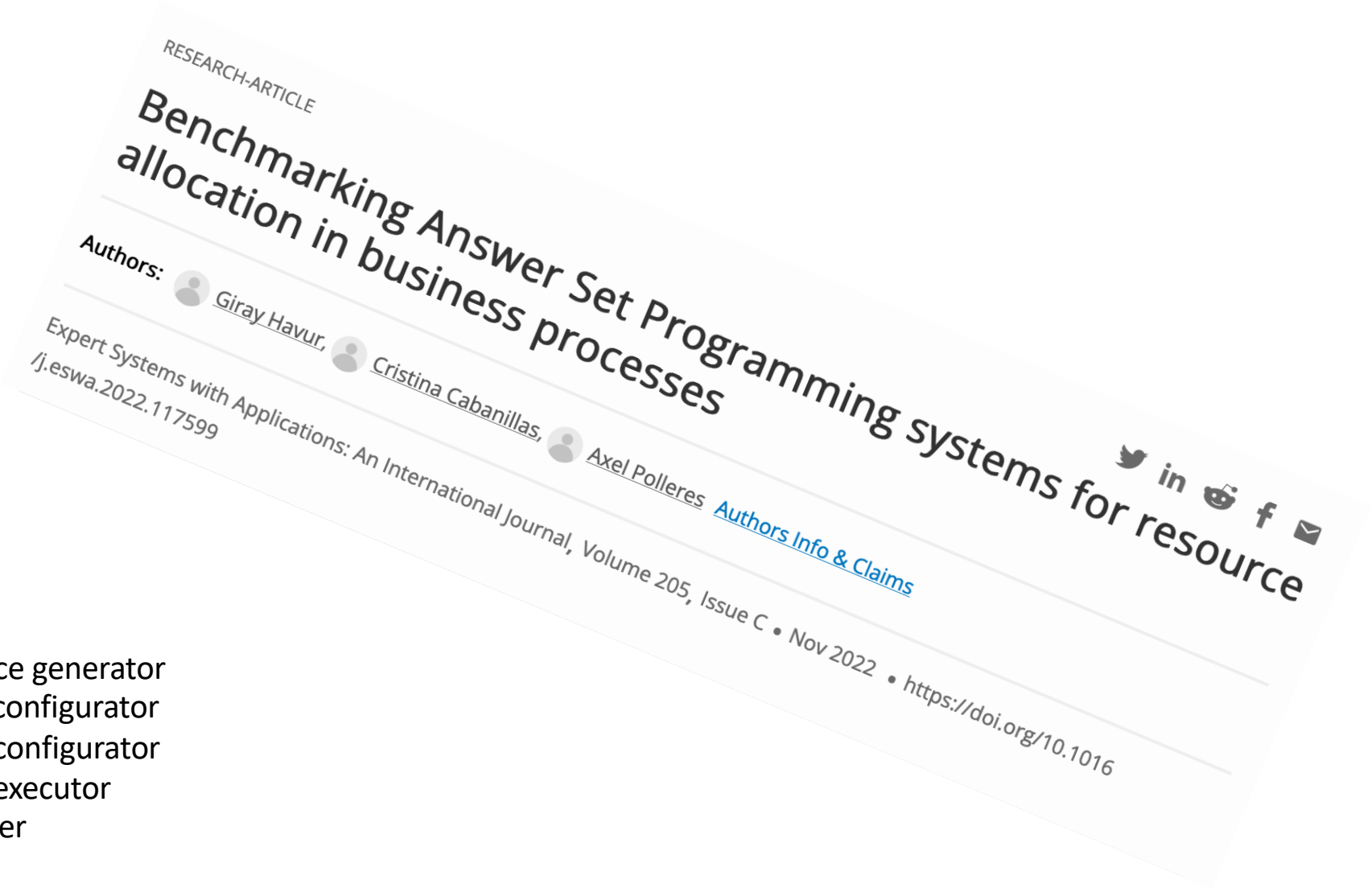




Saimir Bala, Giray Havur, Simon Sperl, Simon Steyskal, Alois Haselbock, Jan Mendling, and Axel Polleres. Shapeworks: A BPMS extension for complex process management. In the BPM Demo Track 2016 co-located with BPM 2016, volume 1789 of CEUR Workshop Proceedings, pages 50–55, 2016.

https://www.youtube.com/watch?v=31PxXoQR9n4

BRANCH: An ASP Systems Benchmark for RABP “hot from the press”


RESEARCH-ARTICLE

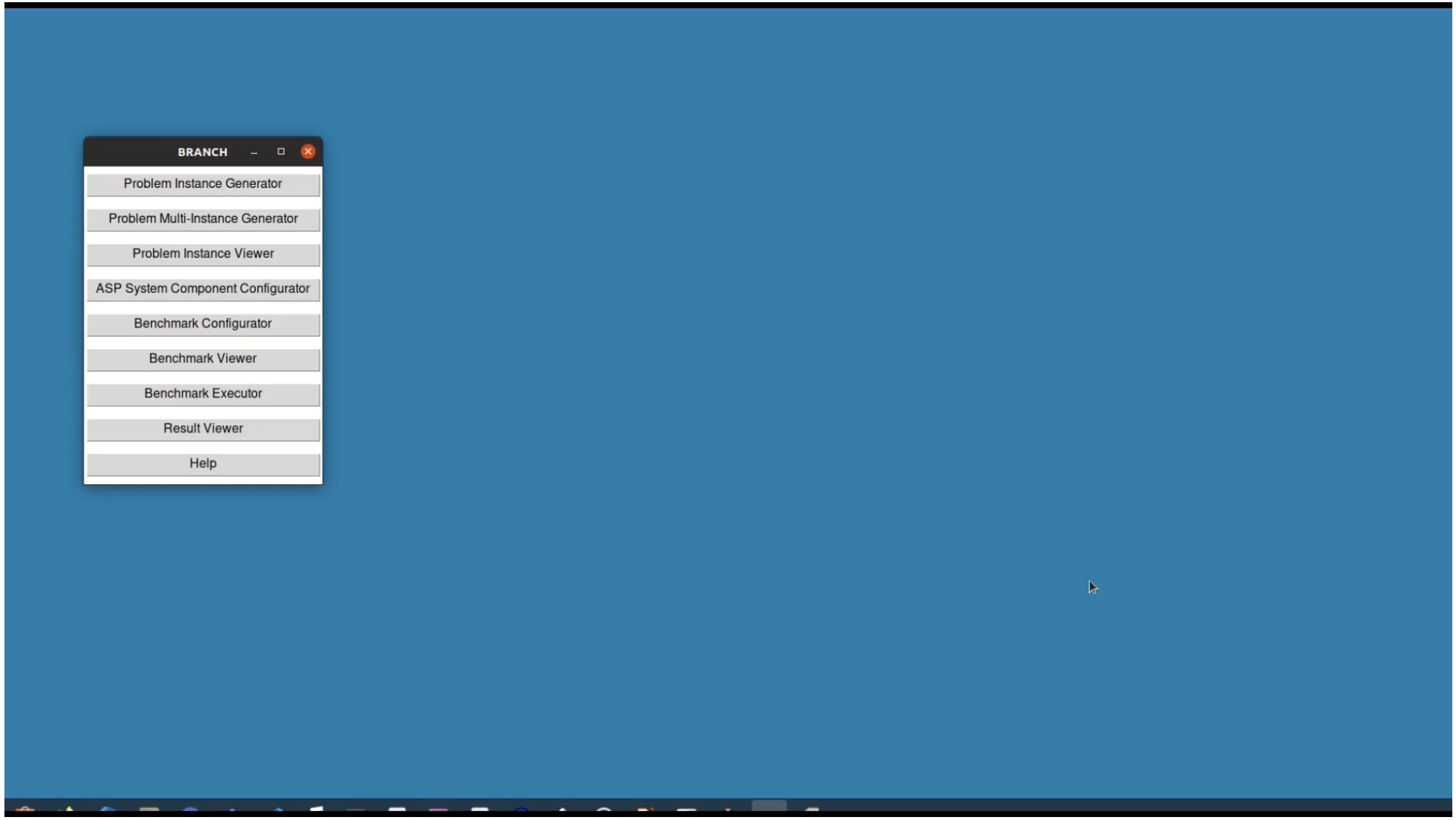
Benchmarking Answer Set Programming systems for resource allocation in business processes

Authors: Giray Havur, Cristina Cabanillas, Axel Polleres Authors Info & Claims

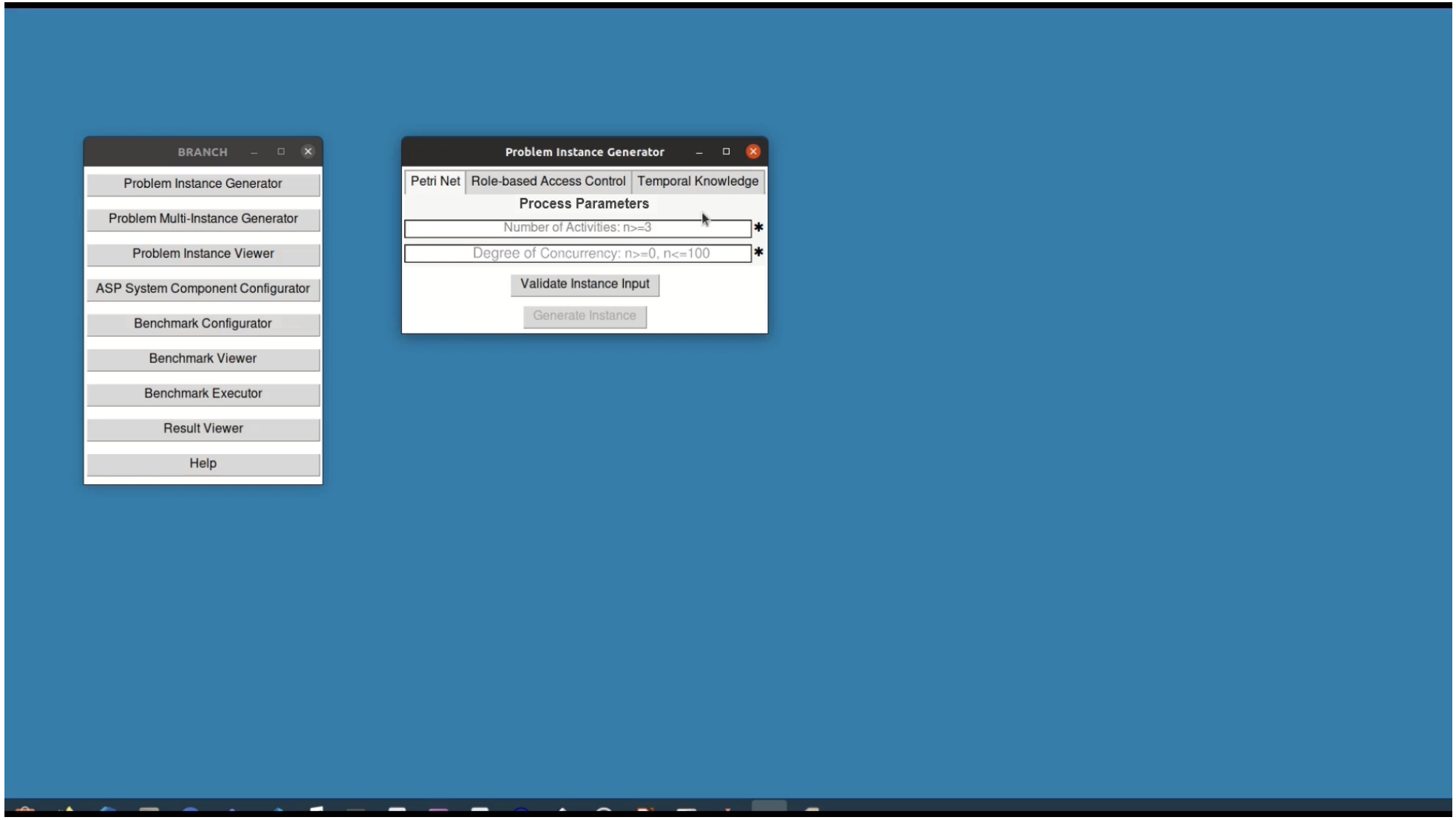
Expert Systems with Applications: An International Journal, Volume 205, Issue C • Nov 2022 • https://doi.org/10.1016/j.eswa.2022.117599

- **BRANCH**
  - RABP instance generator
  - ASP system configurator
  - Benchmark configurator
  - Benchmark executor
  - Results viewer

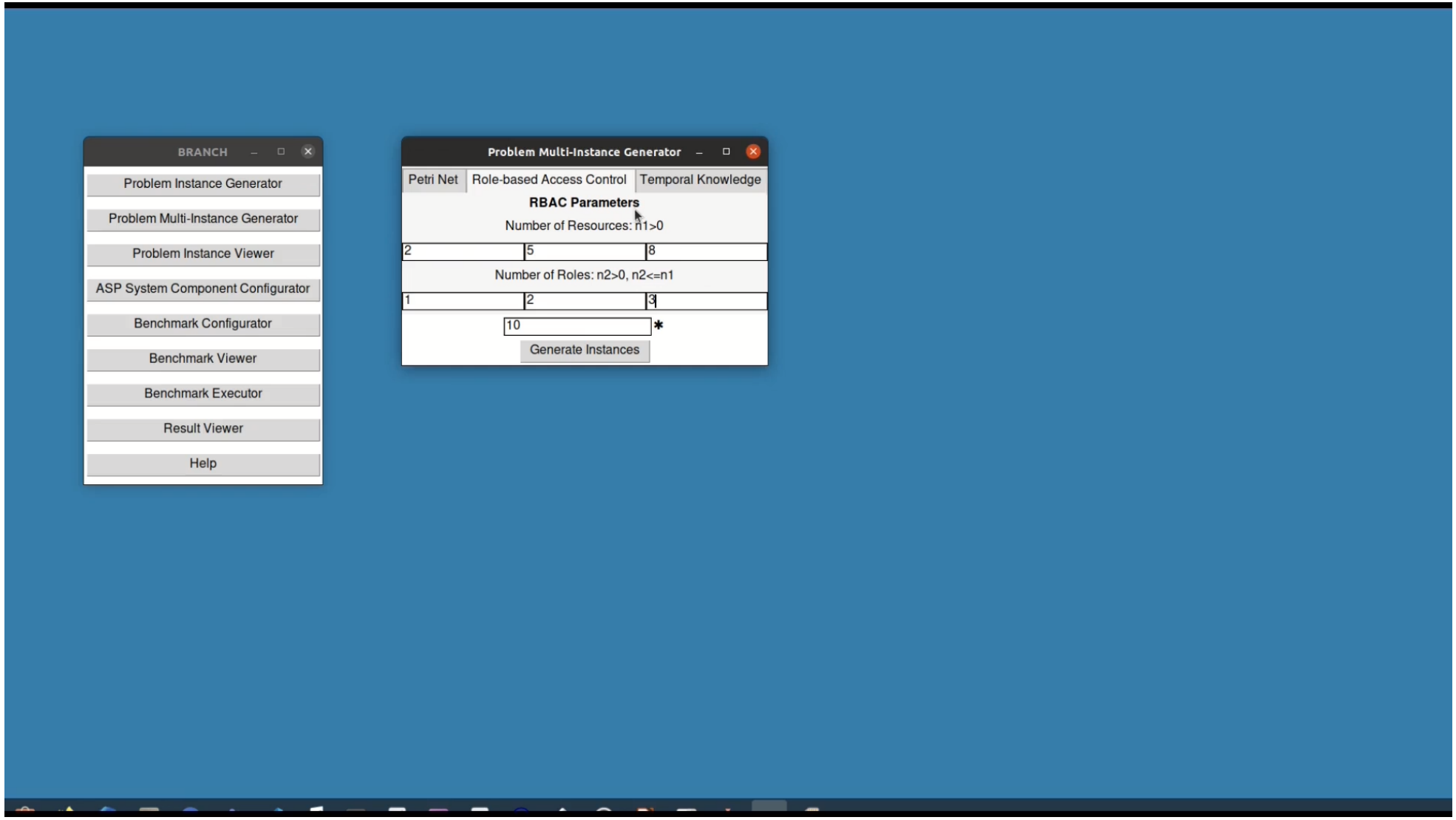
An ASP Systems Benchmark for RABP



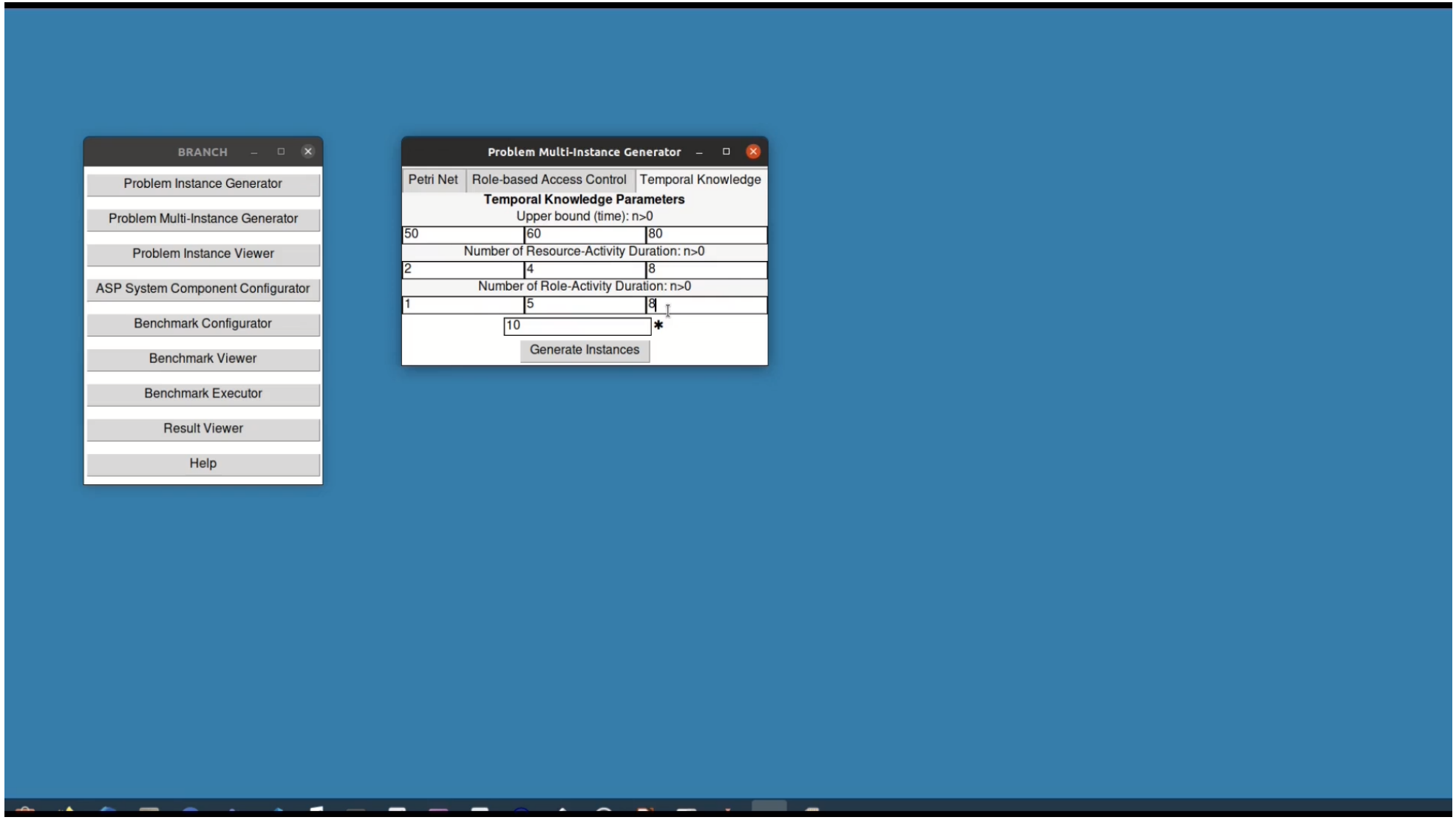
An ASP Systems Benchmark for RABP



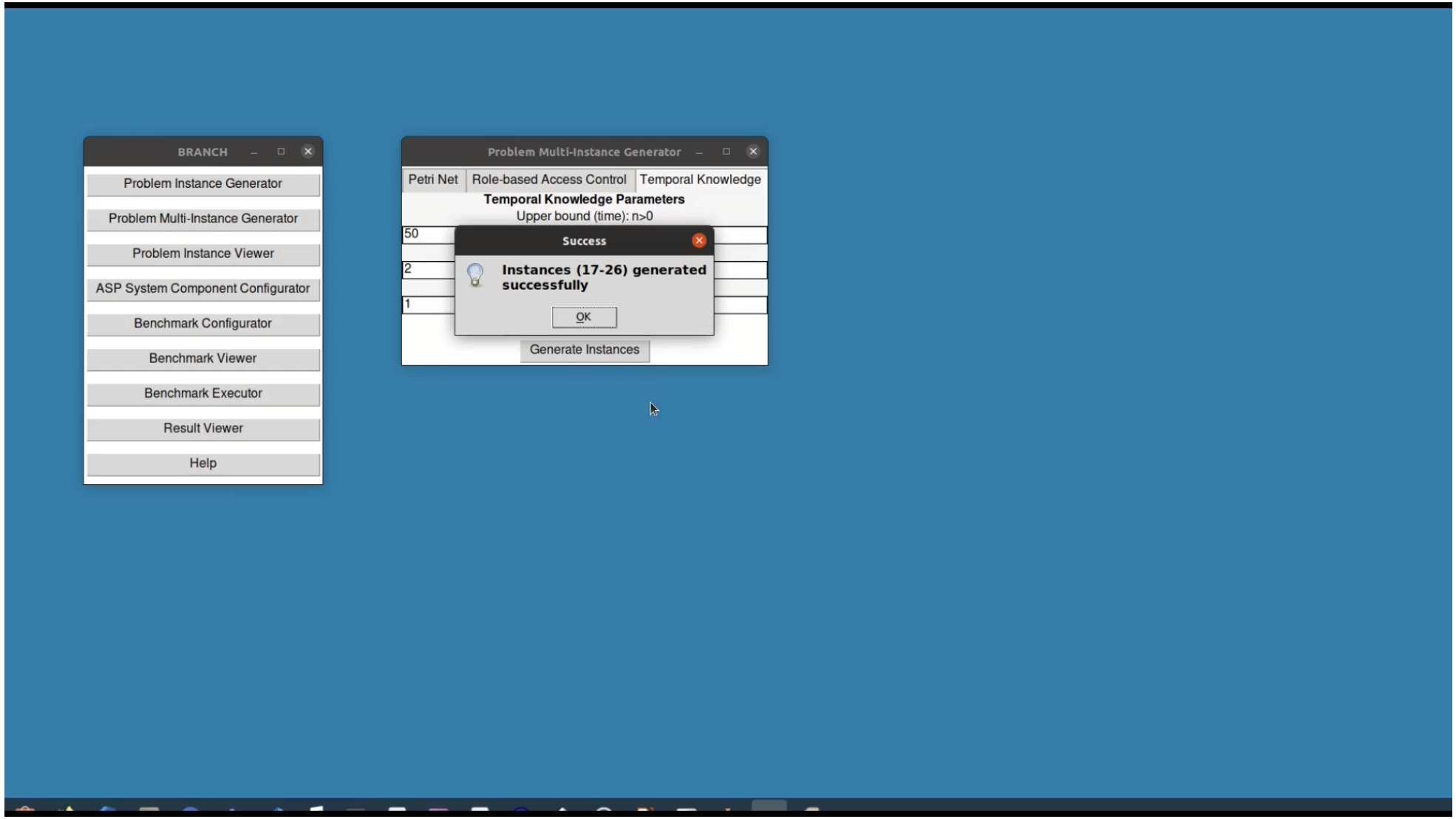
An ASP Systems Benchmark for RABP



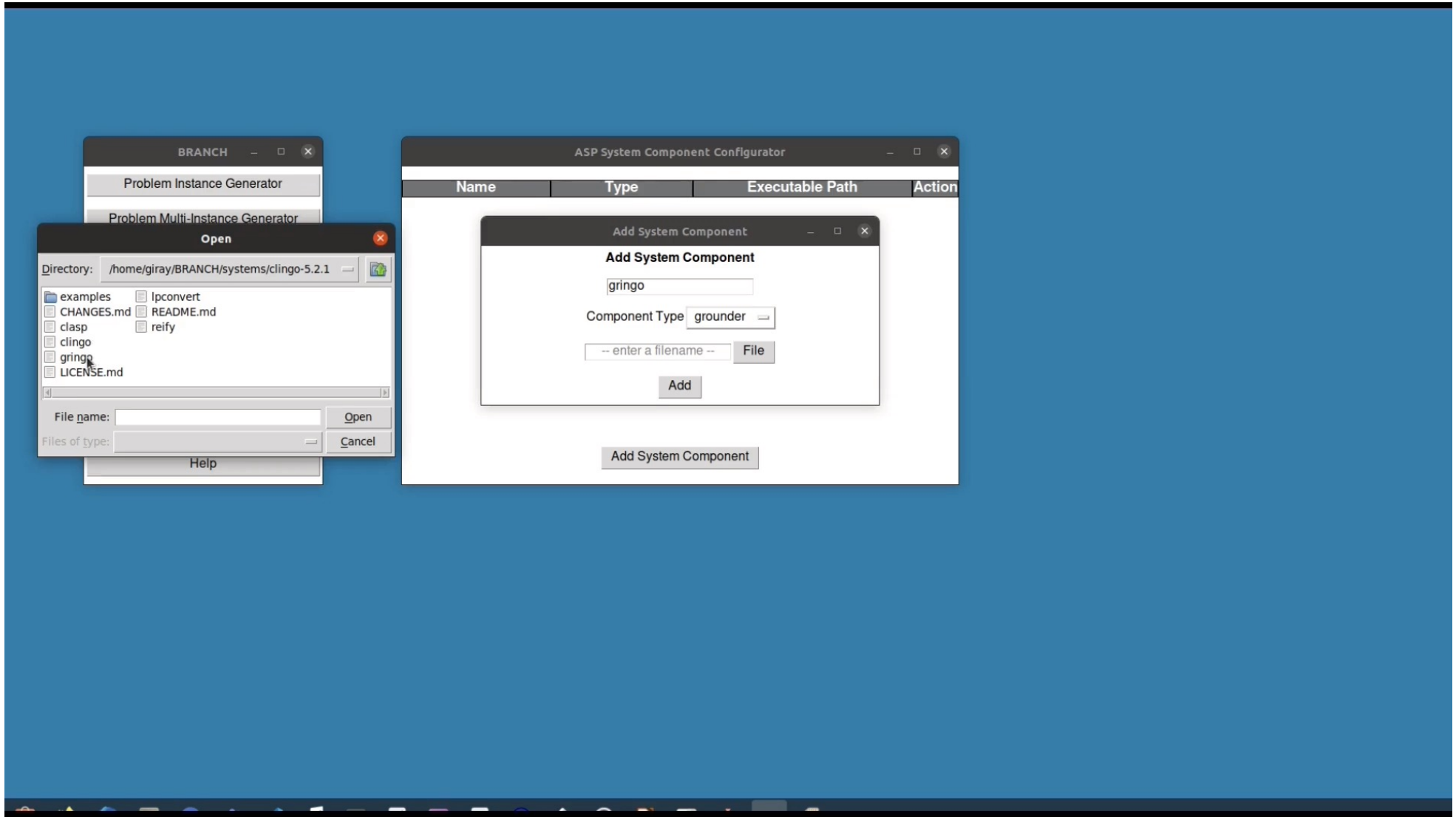
An ASP Systems Benchmark for RABP



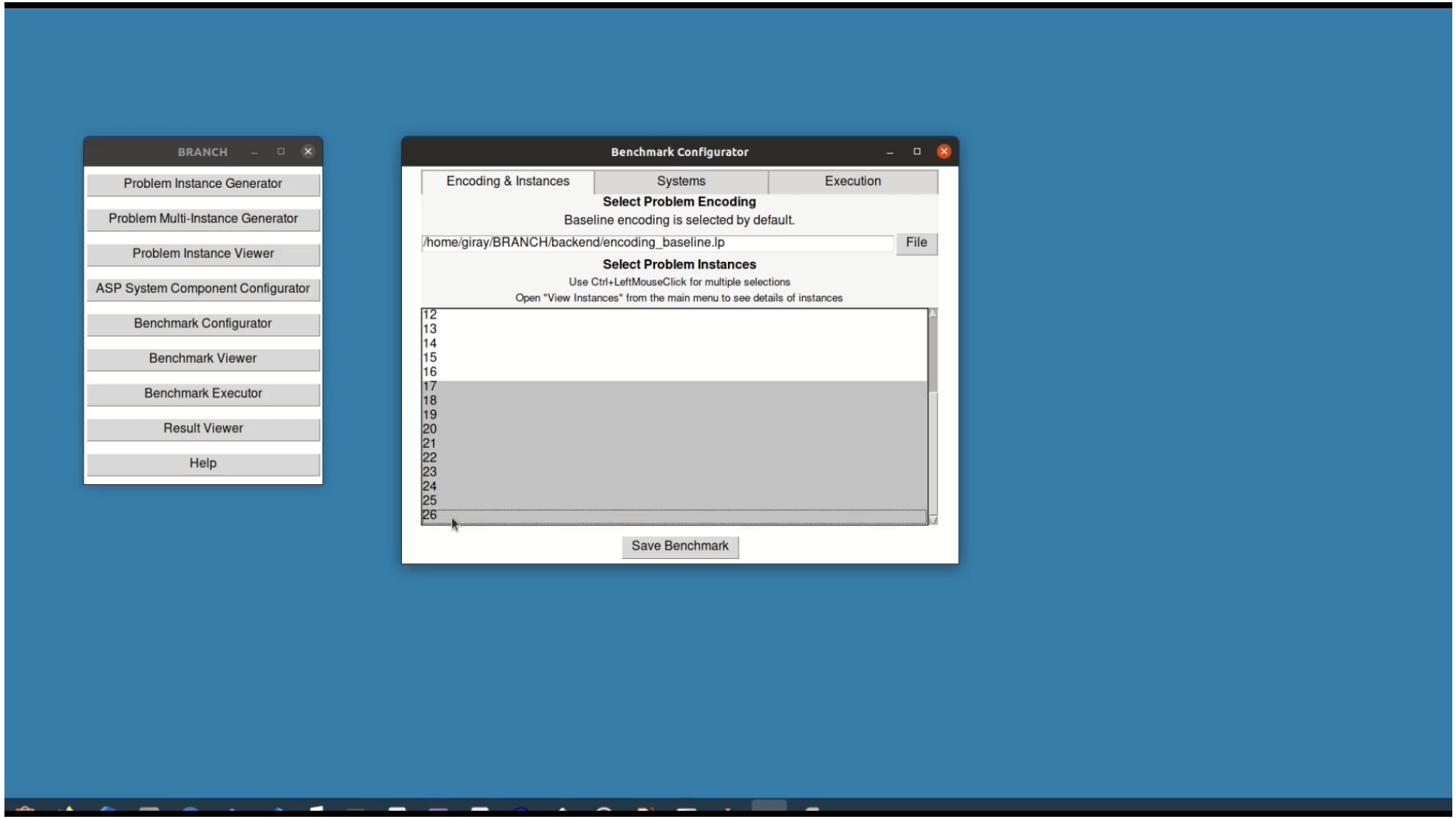
An ASP Systems Benchmark for RABP



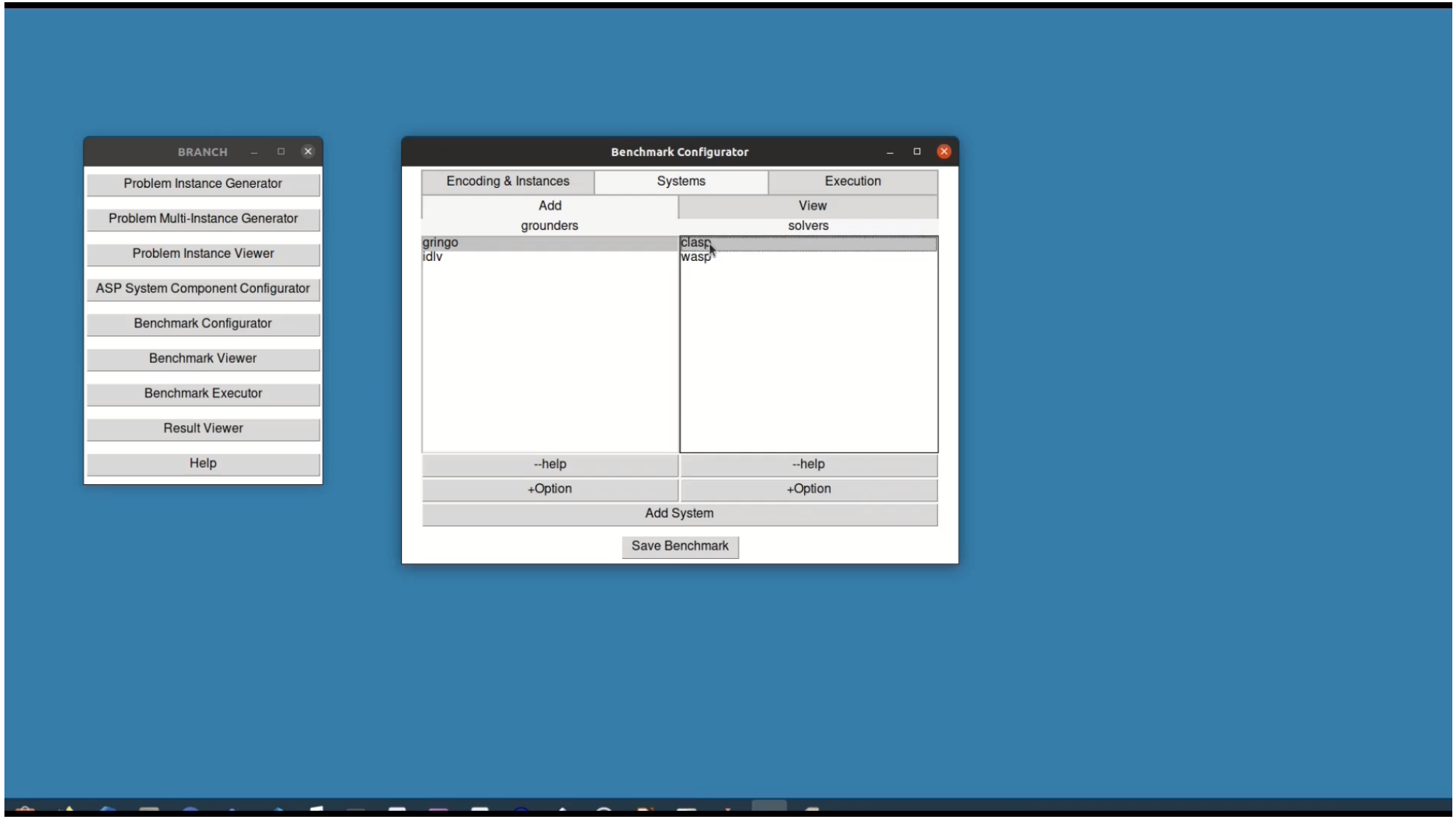
An ASP Systems Benchmark for RABP



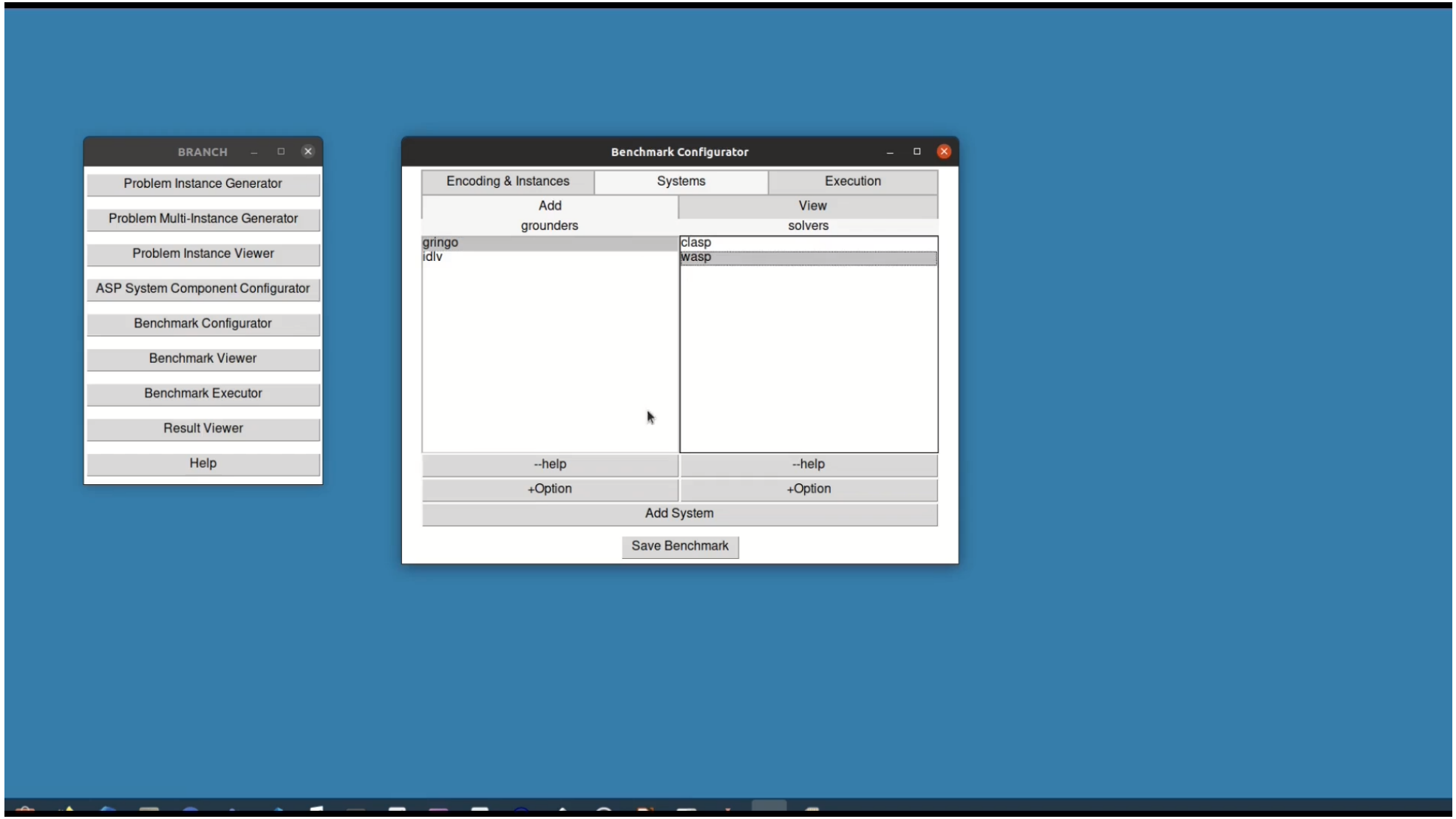
An ASP Systems Benchmark for RABP



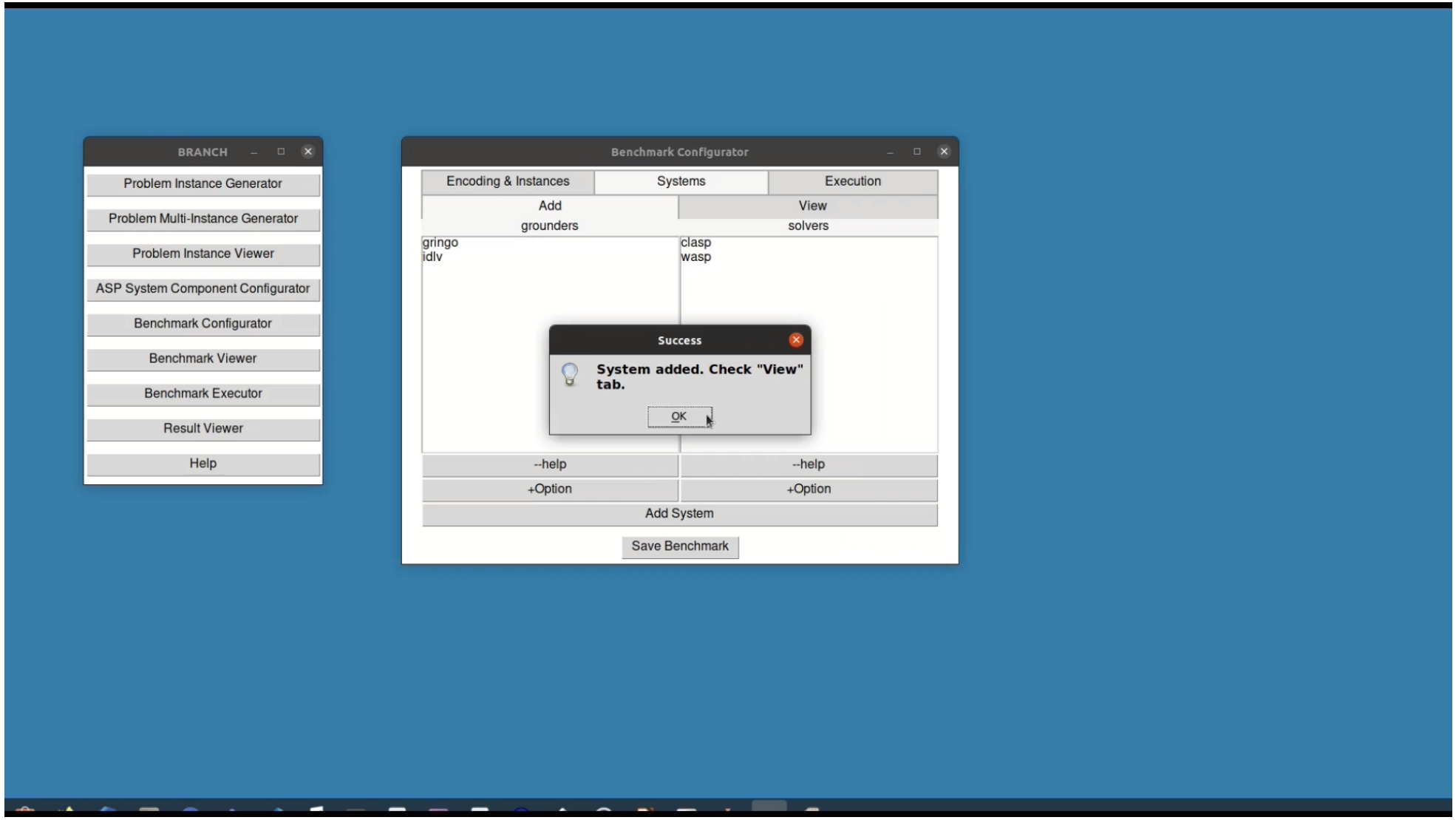
An ASP Systems Benchmark for RABP



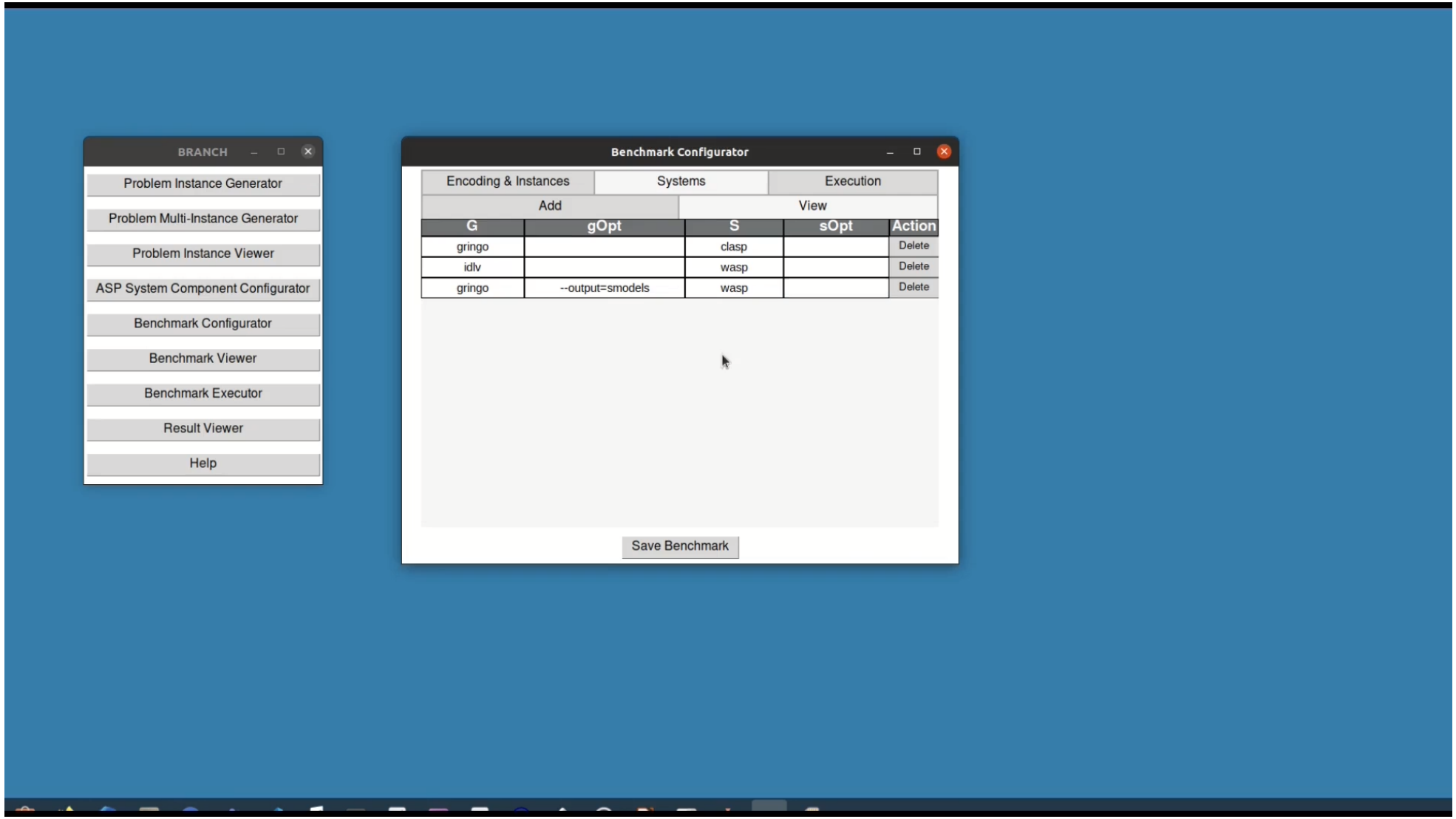
An ASP Systems Benchmark for RABP



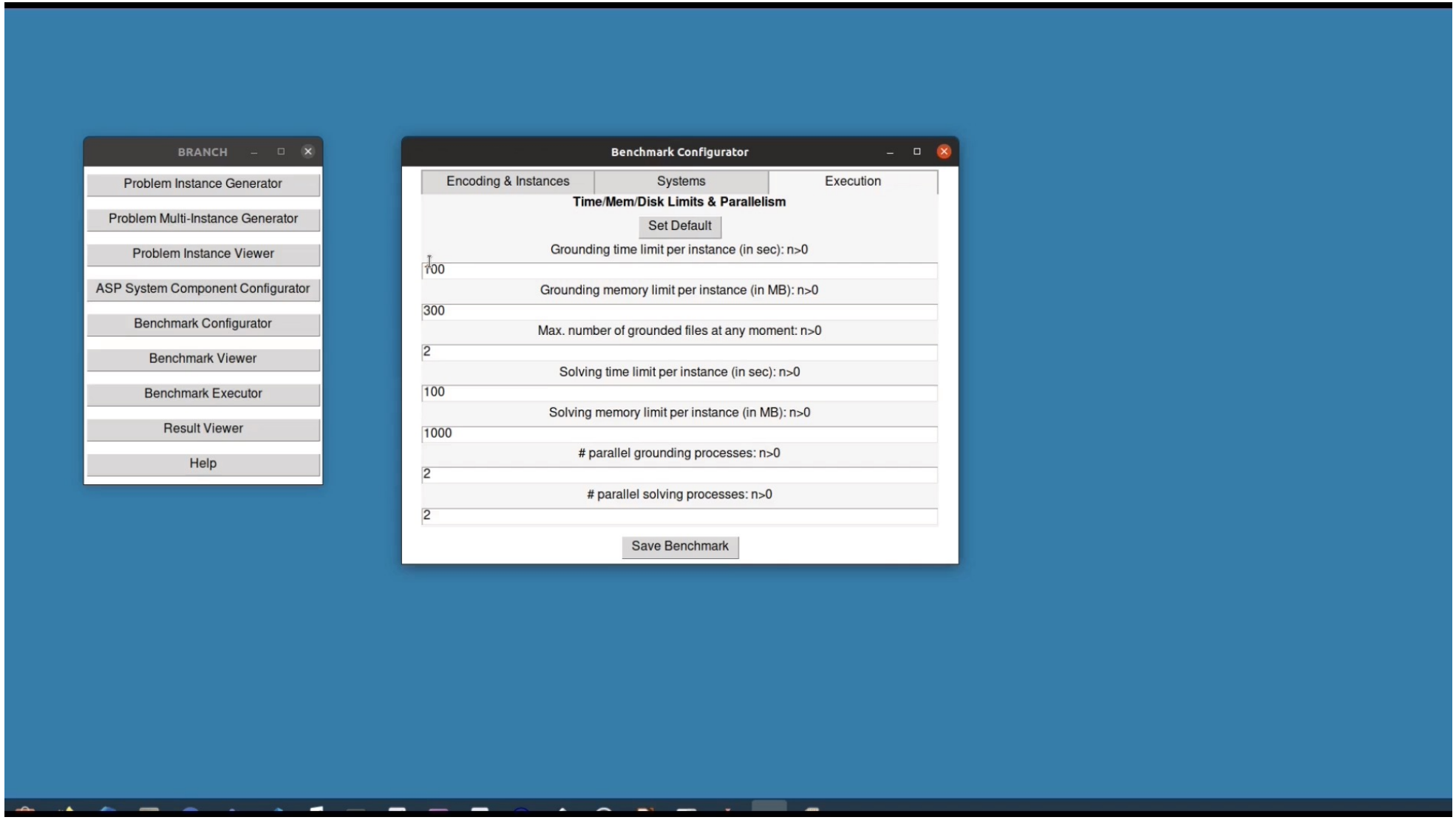
An ASP Systems Benchmark for RABP



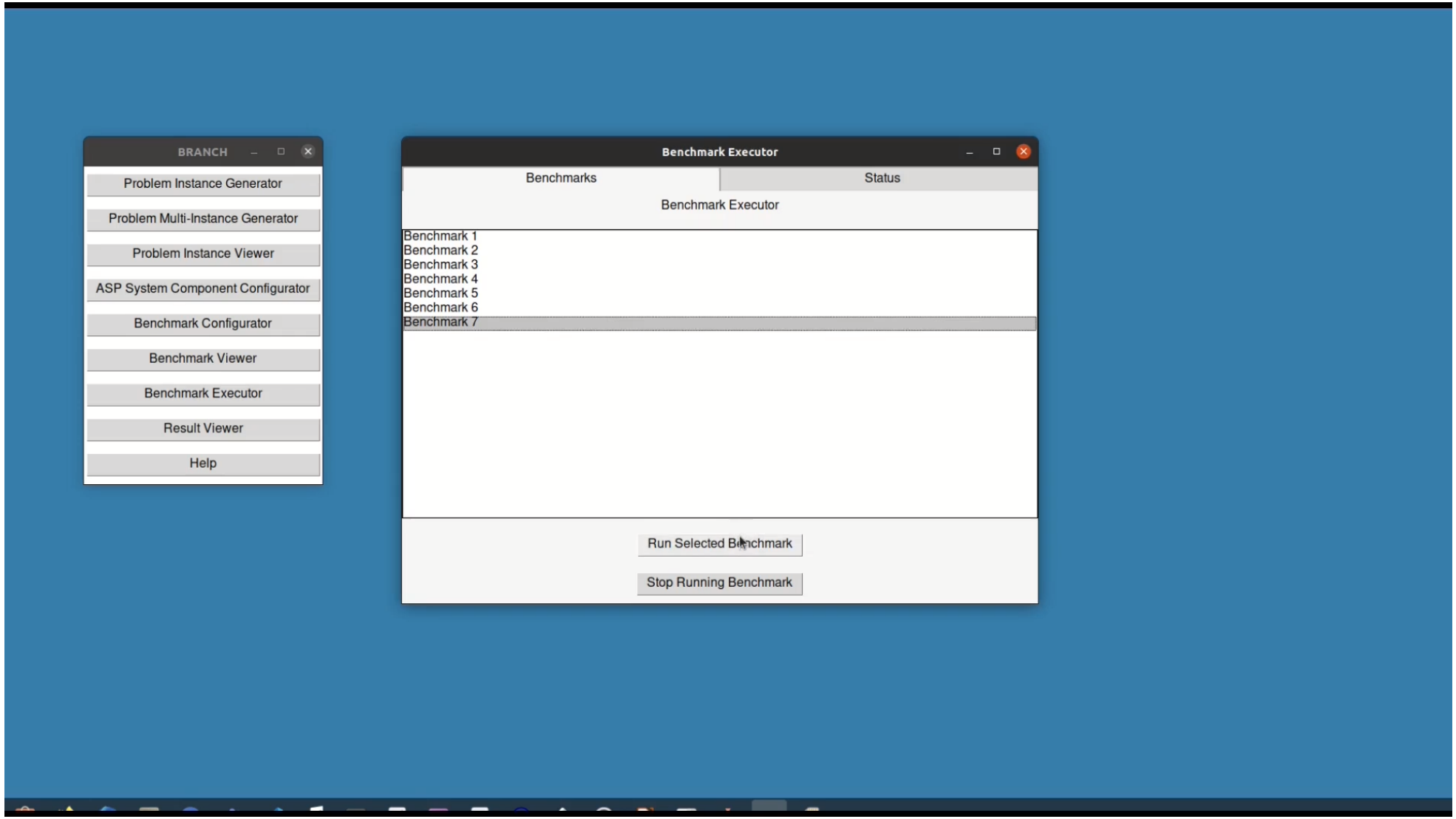
An ASP Systems Benchmark for RABP



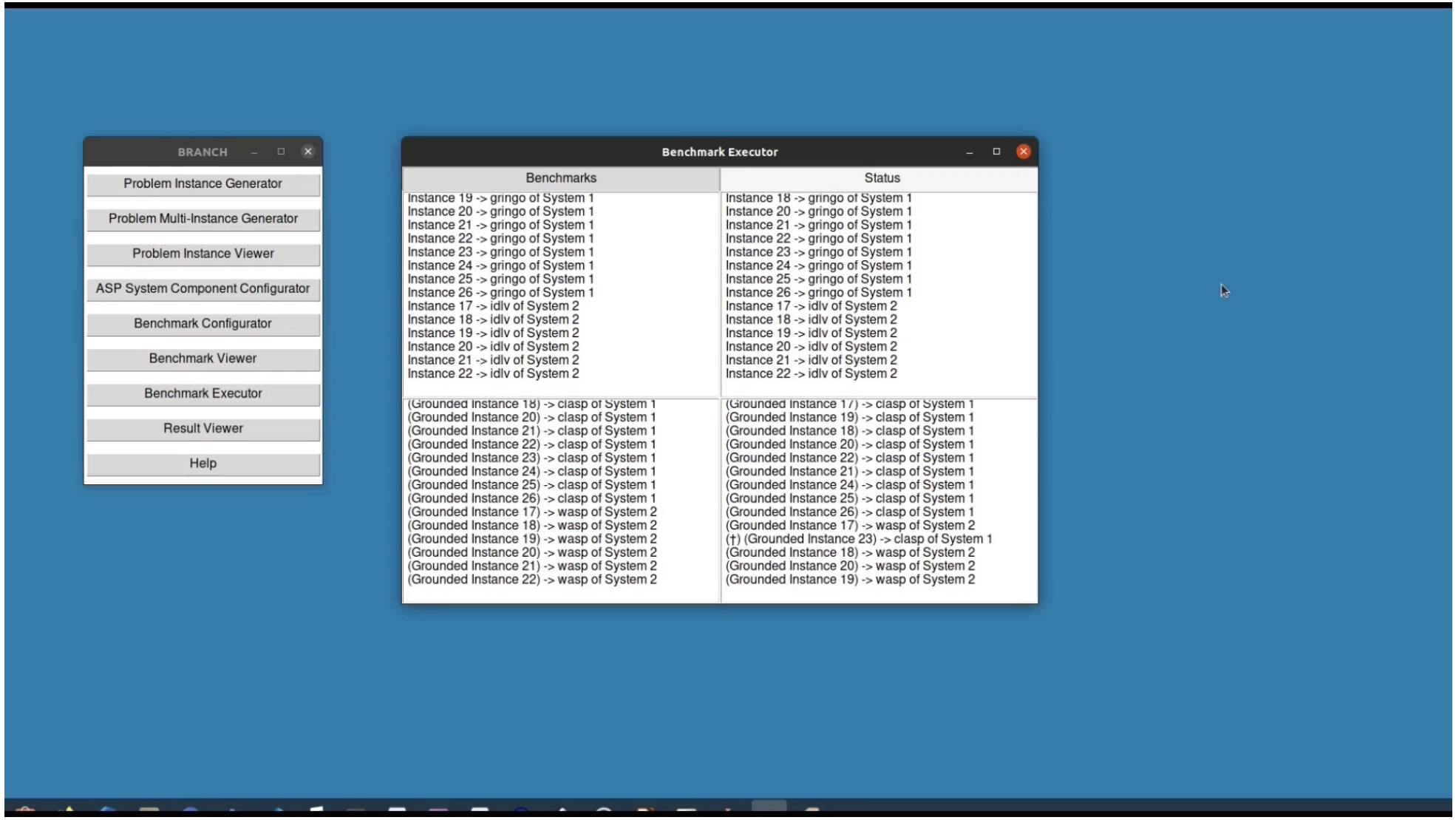
An ASP Systems Benchmark for RABP



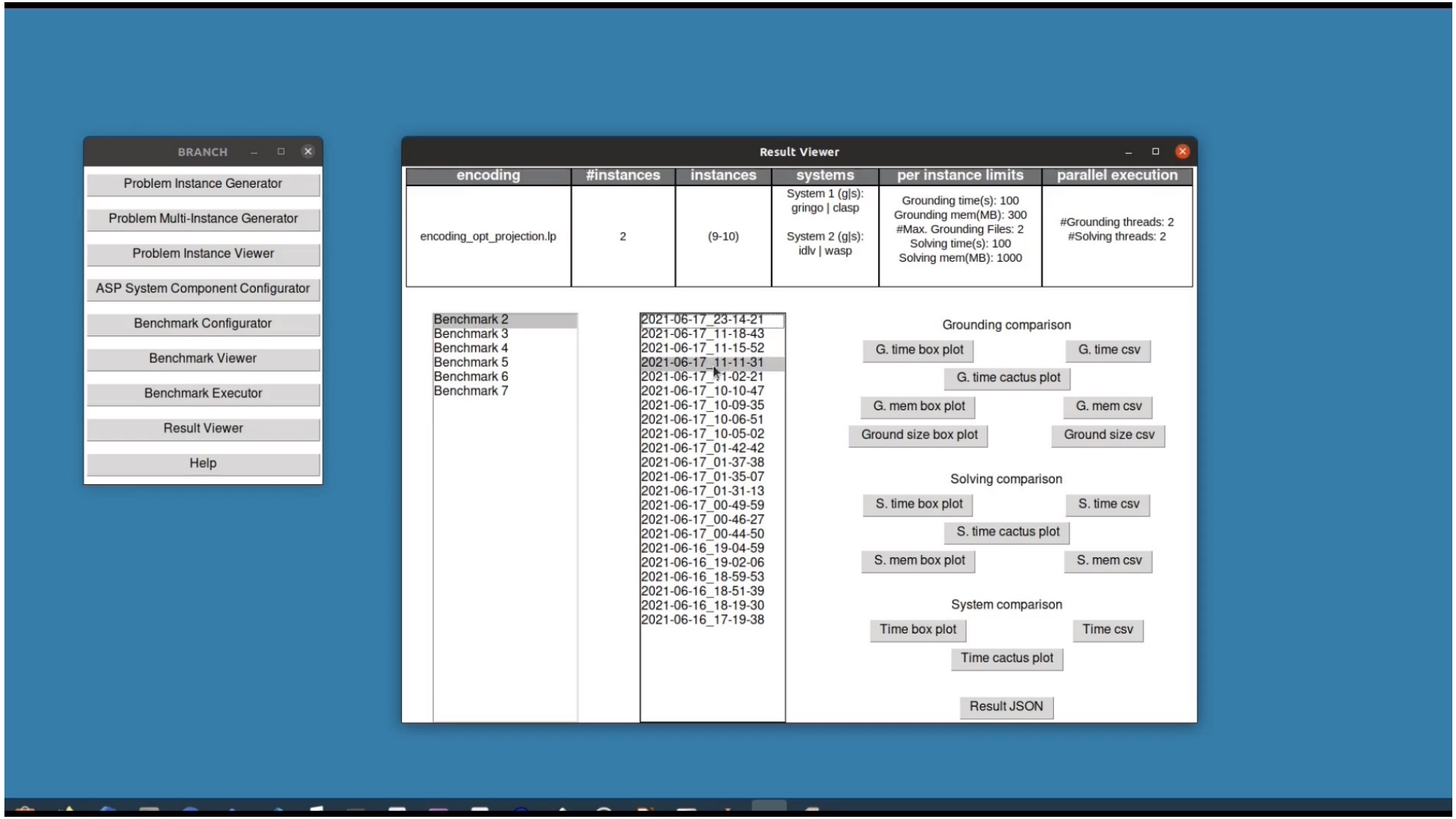
An ASP Systems Benchmark for RABP



An ASP Systems Benchmark for RABP

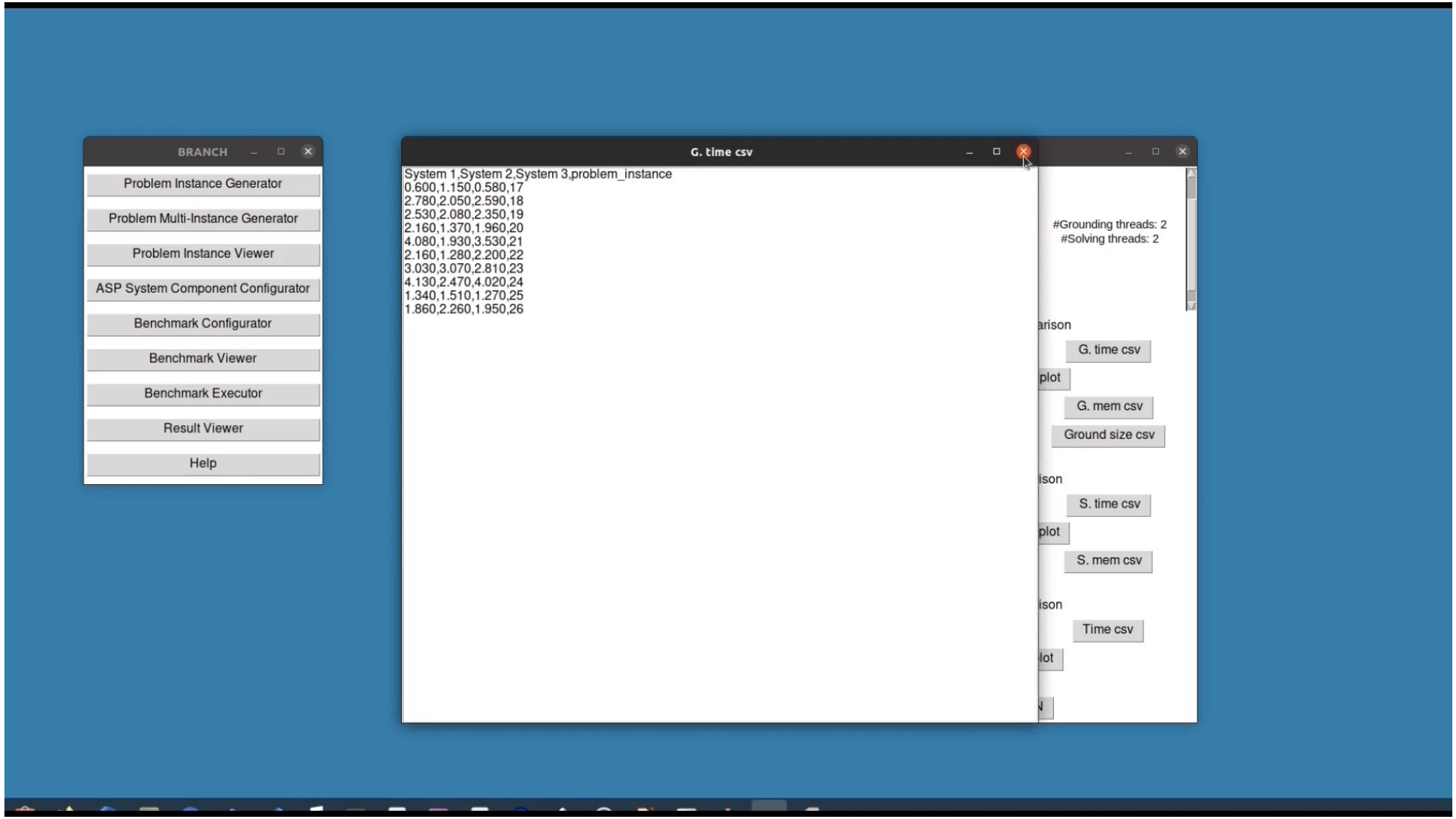


An ASP Systems Benchmark for RABP



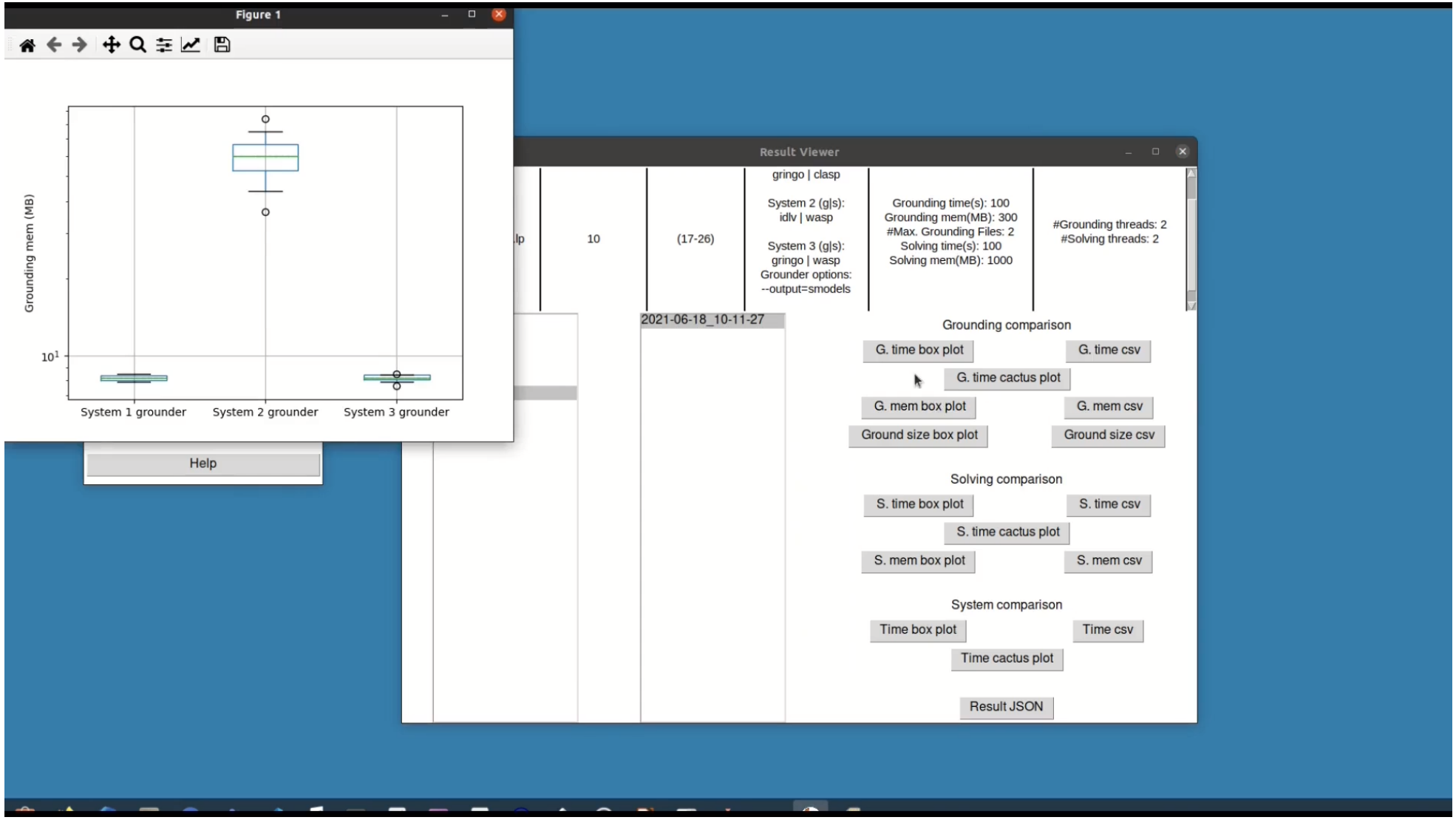
Contributions to RQ 2

An ASP Systems Benchmark for RABP



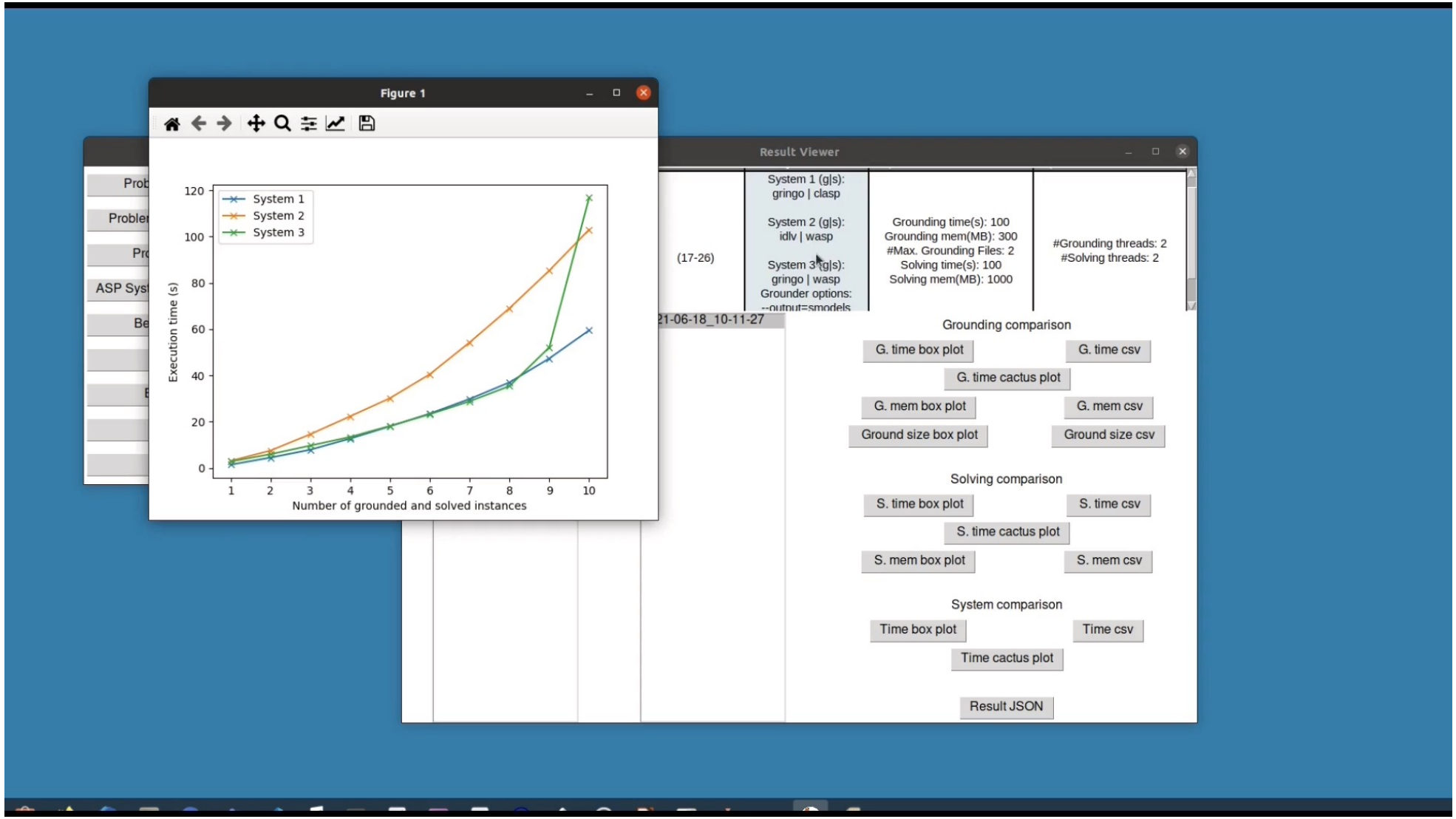
Contributions to RQ 2

An ASP Systems Benchmark for RABP



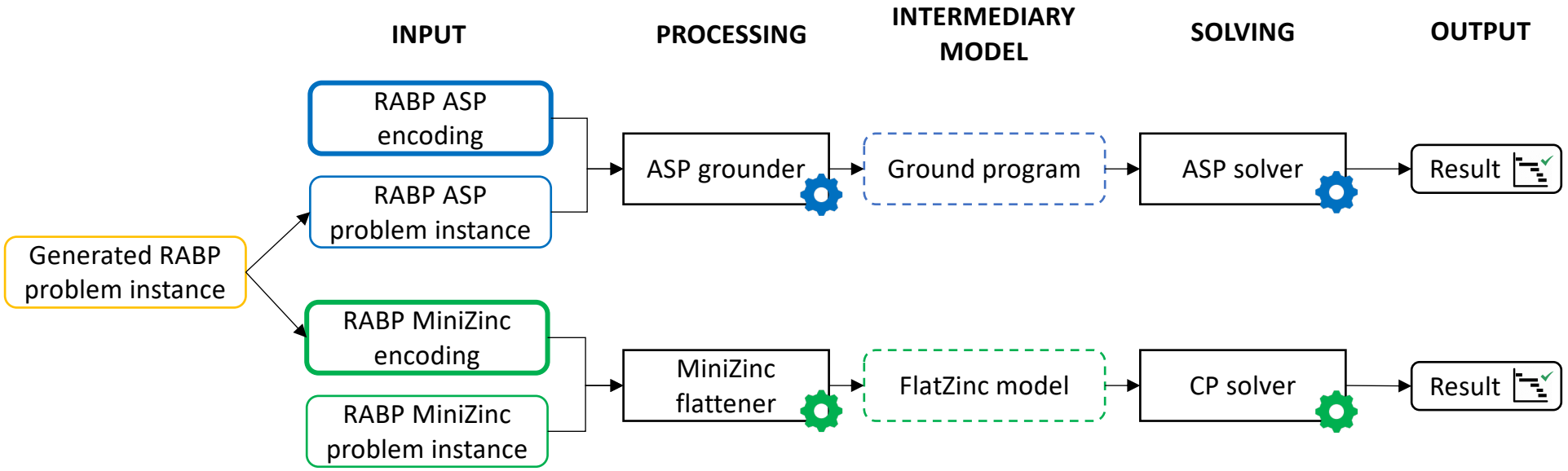
Contributions to RQ 2

An ASP Systems Benchmark for RABP



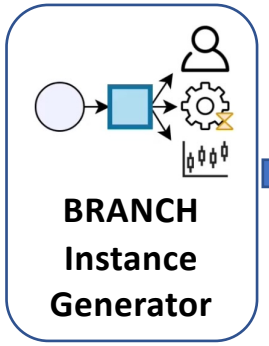
ASP vs CP: Performance Evaluation

- Running RABP Instances in ASP and CP



- Comparison of four ASP systems against four CP systems
  - ASP systems: **gringo+clasp**, **gringo+wasp**, **idlv+clasp**, **idlv+wasp**
  - CP systems: mzn2fzn+**gecode**, mzn2fzn+**chuffed**, mzn2fzn+**hcsp**, mzn2fzn+**or-tools**

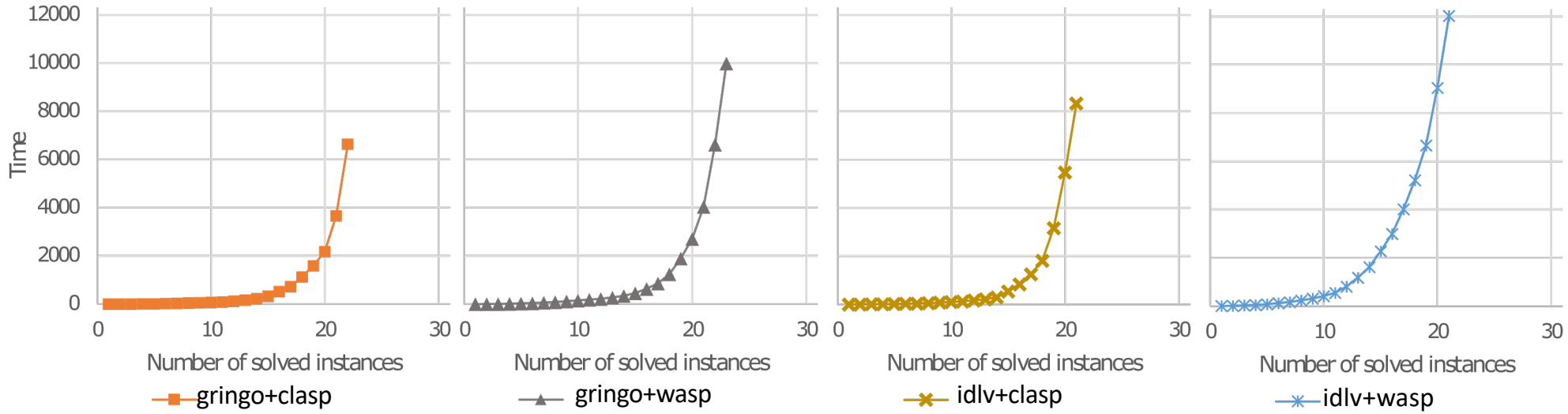
ASP vs CP: Performance Evaluation



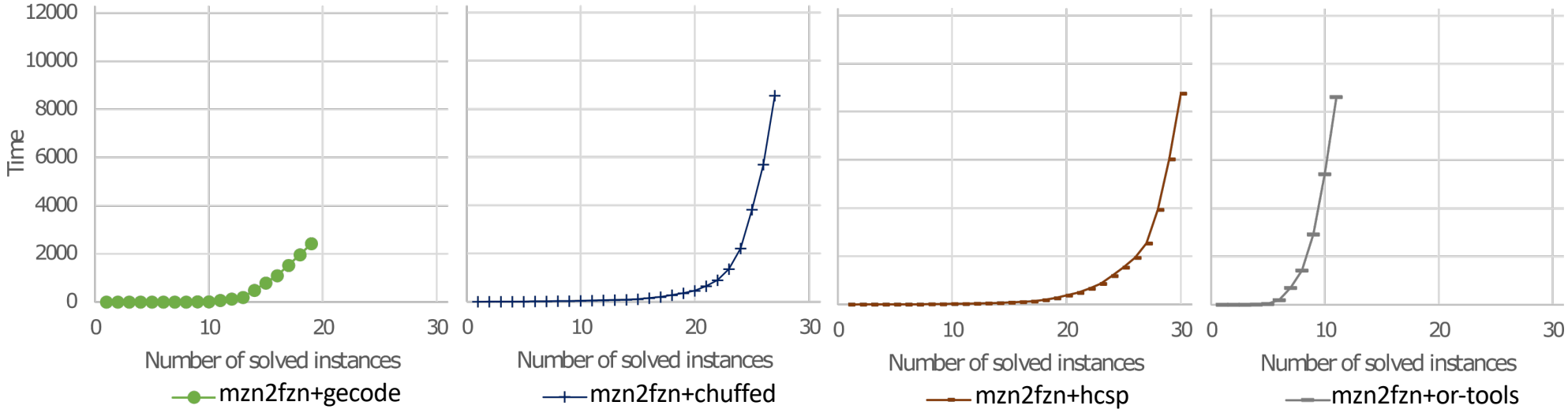
id	#A	%conc	#R	#L	u	id	#A	%conc	#R	#L	u
1	16	10	2	1	32	**22**	16	50	2	1	48
2	16	50	2	1	32	**23**	16	50	2	1	64
3	16	90	2	1	32	**24**	24	50	3	1	72
4	24	10	3	1	48	**25**	24	50	3	1	96
5	24	50	3	1	48	**26**	32	50	4	2	96
6	24	90	3	1	48	**27**	32	50	4	2	128
7	32	10	4	2	64	**28**	48	50	6	3	144
8	32	50	4	2	64	**29**	48	50	6	3	192
9	32	90	4	2	64	**30**	64	50	8	4	192
10	48	10	6	3	96	**31**	64	50	8	4	256
11	48	50	6	3	96	**32**	96	50	12	6	288
12	48	90	6	3	96	**33**	96	50	12	6	384
13	64	10	8	4	128	**34**	128	50	16	8	384
14	64	50	8	4	128	**35**	128	50	16	8	512
15	64	90	8	4	128						
16	96	10	12	6	192						
17	96	50	12	6	192						
18	96	90	12	6	192						
19	128	10	16	8	256						
20	128	50	16	8	256						
21	128	90	16	8	256						

ASP vs CP: Performance Evaluation

ASP Systems



CP Systems



ASP vs CP

	ASP	CP
Performance	++	+++
Readability	+++	+
Encoding RABP Problem	+++	++
Encoding RABP Instances	+++	+

Representing a wide variety of resources in RABP

BPI15: Giray Havur, Cristina Cabanillas, Jan Mendling, and Axel Polleres. Automated resource allocation in business processes with answer set programming. In Business Process Management Workshops: (BPI 2015), Revised Papers, pages 191–203, 2016.

SIMPDA15: Saimir Bala, Cristina Cabanillas, Alois Haselbock, Giray Havur, Jan Mendling, Axel Polleres, Simon Sperl, and Simon Steyskal. A framework for safety-critical process management in engineering projects. In SIMPDA 2015, Revised Selected Papers, volume 244 of LNBIP, pages 1–27, Springer, 2015.

BPMDemo15: Saimir Bala, Giray Havur, Simon Sperl, Simon Steyskal, Alois Haselbock, Jan Mendling, and Axel Polleres. Shapeworks: A BPMS extension for complex process management. In the BPM Demo Track 2016 co-located with BPM 2016, volume 1789 of CEUR Workshop Proceedings, pages 50–55, 2016.

BPMForum16: Giray Havur, Cristina Cabanillas, Jan Mendling, and Axel Polleres. Resource allocation with dependencies in business process management systems. In Business Process Management Forum, pages 3–19, 2016.

RESEARCH-ARTICLE

Benchmarking Answer Set Programming systems for resource allocation in business processes

Authors: Giray Havur, Cristina Cabanillas, Axel Polleres Authors Info & Claims

Expert Systems with Applications: An International Journal, Volume 205, Issue C • Nov 2022 • https://doi.org/10.1016/j.eswa.2022.117599

More details in Giray's thesis…

Thank you!

Why I love ASP (since over 20 years)…

- ***Intuitive***, understandable problem ***encodings***…`
- … easily ***extensible***
- the beauty of **Guess and Check** to solve complex problems on top
- integrateable in real systems… looking forward to learn more from you!

Part 1:
1999-2003

AI Planning

Part 2:
2003 – to date…

Universidad
Rey Juan Carlos

Semantic Web

Part 3:
ca. 2014 – 2022…

Business Process
Management