

A System for Dispute Mediation: The Mediation Dialogue Game

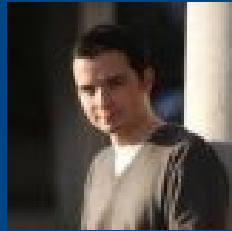
Mathilde Janier, Mark Snaith, Katarzyna Budzynska
John Lawrence & Chris Reed

Centre for Argument Technology
University of Dundee & Polish Academy of Sciences

<http://arg.tech>



Mathilde
Janier



Mark
Snaith



Kasia
Budzynska



John
Lawrence



The Leverhulme Trust



List of all technology tools for supporting mediators

Dialogue games



Dialogue games



Dialogue games

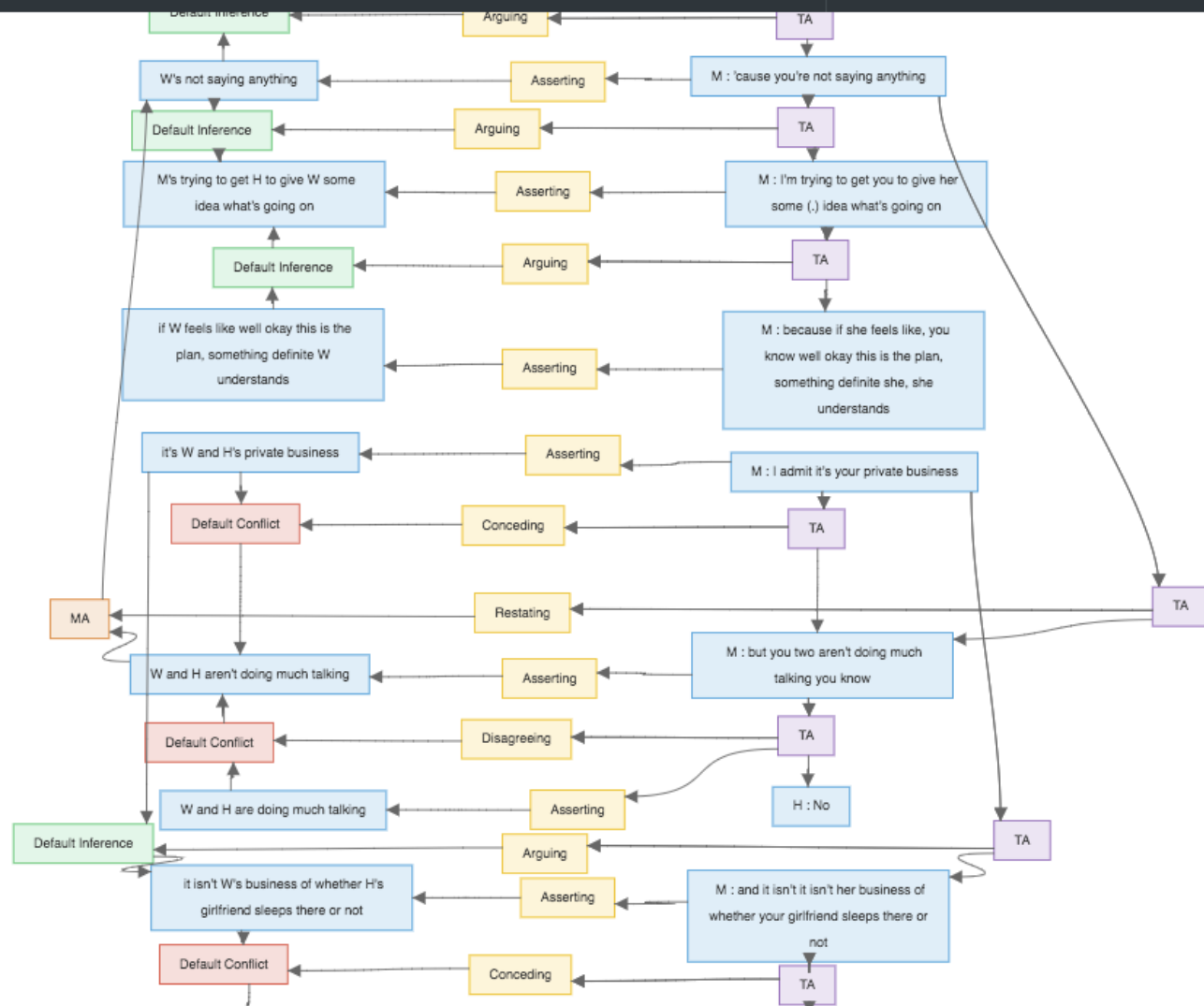


Dialogue games



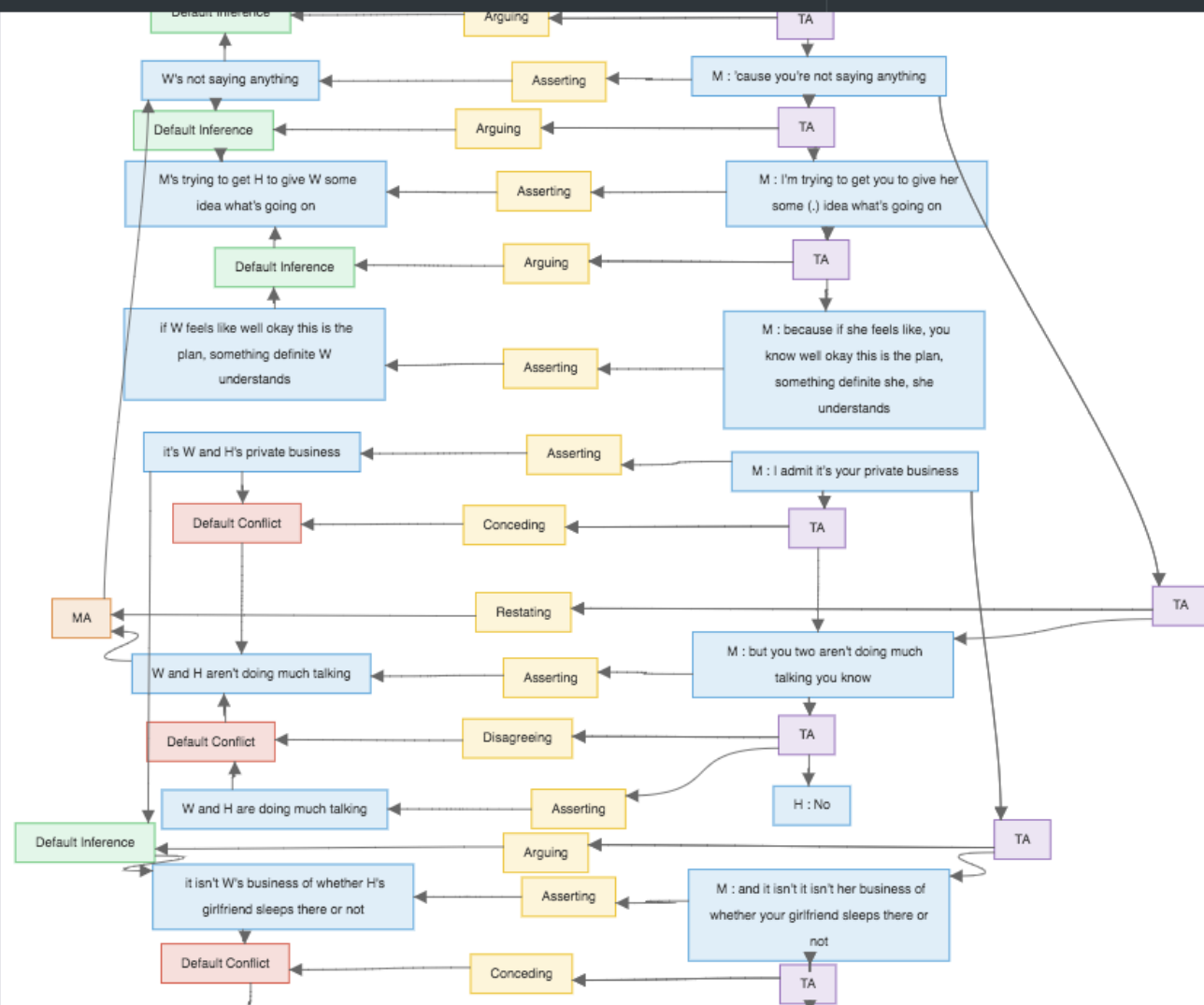
Data driven

((pause)) 013 M: Well if your girlfriend sleeps overnight then where is Jason going to sleep 014 H: Hmm 015 M: What if your girlfriend sleeps overnight with you then where is Jason gonna sleep 016 H: Uh they'll still sleep with me I guess 1408 S. Jacobs / Journal of Pragmatics 34 (2002) 1403-1426 017 M: I don't think it's a good idea, having him sleep between you and your girlfriend. (.) that wouldn't be, that wouldn't be [a good plan you ought to have another] room for [him] 018 H: [Well I plan I'm gonna (.)] [Now] listen, Jason is my son. What I do with my son I think is my business, not yours of course or anybody else's [as long as he is taken care of, he's, well who [does] but as long as, he's my son too] [she should (.)] (go ahead)= 019 M: [his mother's] [But (.)], I don't care =but I'm talking sort of talking for her (.) I'm trying 'cause you're not saying anything I'm trying to get you to give her some (.) idea what's going on because if she feels like, you know well okay this is the plan, something definite she, she understands (.) I admit it's your private business, but you two aren't doing much talking you know 020 H: No, I ah (.) 021 M: [A, a,] and it isn't it isn't her business of whether your girlfriend sleeps there or not! but it is her business, if the girlfriend sleeps there and Jason's there (.) you know 022 H: U[h (.) her boyfriend] sleeps at her [house] (.) 023 M: [Just like it's your business] [Yeah] you have a [right] to ask is does does Jason sleep between you and your boyfriend or is Jason in his crib, and is the crib in the same room or what, you know, that's fine you you have a right to, to ask that because that's not a proper atmosphere I wouldn't think for (.) [for] your little, for your child, (.) you know= 024 H: [Well] [Well] =maybe at that age you know, what is he gonna know, you know I can see, maybe sleeping with you when he gets older and he knows what's going on



Data driven

((pause)) 013 M: Well if your girlfriend sleeps overnight then where is Jason going to sleep 014 H: Hmm 015 M: What if your girlfriend sleeps overnight with you then where is Jason gonna sleep 016 H: Uh they'll still sleep with me I guess 1408 S. Jacobs / Journal of Pragmatics 34 (2002) 1403-1426 017 M: I don't think it's a good idea, having him sleep between you and your girlfriend. (.) that wouldn't be, that wouldn't be [a good plan you ought to have another] room for [him] 018 H: [Well I plan I'm gonna (.)] [Now] listen, Jason is my son. What I do with my son I think is my business, not yours of course or anybody else's [as long as he is taken care of, he's, well who [does] but as long as, he's my son too] [she should (.)] (go ahead)= 019 M: [his mother's] [But (.)], I don't care =but I'm talking sort of talking for her (.) I'm trying 'cause you're not saying anything I'm trying to get you to give her some (.) idea what's going on because if she feels like, you know well okay this is the plan, something definite she, she understands (.) I admit it's your private business, but you two aren't doing much talking you know 020 H: No, I ah (.) 021 M: [A, a,] and it isn't it isn't her business of whether your girlfriend sleeps there or not! but it is her business, if the girlfriend sleeps there and Jason's there (.) you know 022 H: U[h (.) her boyfriend] sleeps at her [house] (.) 023 M: [Just like it's your business] [Yeah] you have a [right] to ask is does does Jason sleep between you and your boyfriend or is Jason in his crib, and is the crib in the same room or what, you know, that's fine you you have a right to, to ask that because that's not a proper atmosphere I wouldn't think for (.) [for] your little, for your child, (.) you know= 024 H: [Well] [Well] =maybe at that age you know, what is he gonna know, you know I can see, maybe sleeping with you when he gets older and he knows what's going on



Search

Dispute mediation

Download full corpus: [tar.gz](#) | [zip](#)

Argument Map 9082

that's right Ann's dreaming into this all kinds of wonderful things that she thinks that don't exist at all they weren't wonderful just for your information that's not Ann's idea of a wonderful thing, having Philip put his big fat hand on her knee Philip's never put his hand Ann, Philip and...

Download: [SVG](#) | [PNG](#) | [DOT](#) | [JSON](#) | [LKIF](#) | [RTNL](#) | [RDF](#) | [PL](#) Edit: [OVA](#) | [OVA+](#)

Argument Map 9083

Natalie'd rather the full custody Natalie would you go along with that too Natalie would not go along with that xxx is why Natalie doesn't want Robert to have any custody of the children...

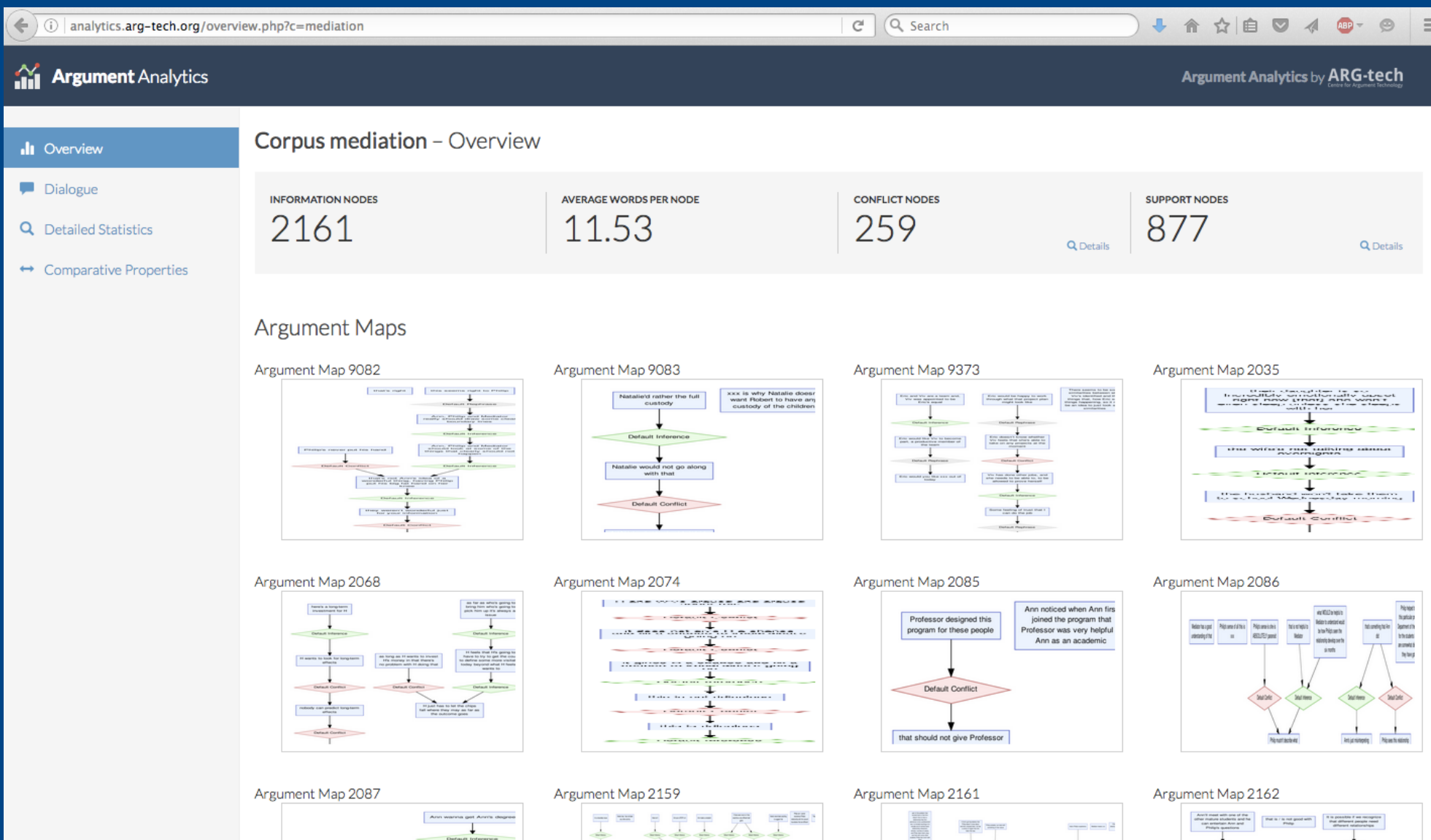
Download: [SVG](#) | [PNG](#) | [DOT](#) | [JSON](#) | [LKIF](#) | [RTNL](#) | [RDF](#) | [PL](#) Edit: [OVA](#) | [OVA+](#)

Argument Map 9373

Eric would like Viv to become part, a productive member of the team Eric would you like xxx out of today Eric and Viv are a team and, Viv was appointed to be Eric's equal Viv would like xxx out of today Some feeling of trust that I can do the job Viv has done other jobs, and she needs to be able to,...

MEDIATION DIALOGUE GAME – JANIER ET AL

11 / 23



Regularities

analytics.arg-tech.org/stats.php?c=mediation

Argument Analytics

Argument Analytics by ARG-tech

mediation – Detailed Statistics

Counts

Item	Count	Item	Count	Item	Count
Asserting	2429	Explaining	72	Rejecting	7
Asserting in Transition	58	Explaining in Transition	72	Rejecting in Transition	7
Asserting in Locution	2369	Explaining in Locution	0	Rejecting in Locution	0
Disagreeing	171	YA	17	Accepting	4
Disagreeing in Transition	171	YA in Transition	9	Accepting in Transition	4
Disagreeing in Locution	0	YA in Locution	8	Accepting in Locution	0
Arguing	761	Rhetorical Questioning	51	Offering	18
Arguing in Transition	761	Rhetorical Questioning in Transition	0	Offering in Transition	0
Arguing in Locution	0	Rhetorical Questioning in Locution	51	Offering in Locution	18
Agreeing	246	Assertive Challenging	7	Default Conflict	260
Agreeing in Transition	246	Assertive Challenging in Transition	1	Default Inference	853
Agreeing in Locution	0	Assertive Challenging in Locution	6	Default Rephrase	274
Assertive Questioning	139	Weak Asserting	3	Reframe	55
Assertive Questioning in Transition	16	Weak Asserting in Transition	0	Example	3
Assertive Questioning in Locution	121	Weak Asserting in Locution	3	Default Paraphrase	33
Restating	283	Directive Questioning	32	Direct Ad Hominem	4
Restating in Transition	283	Directive Questioning in Transition	1	Reframing	2
Restating in Locution	0	Directive Questioning in Locution	31	MA	62
Pure Challenging	22	Ironic Asserting	4	RA	47
Pure Challenging in Transition	9	Ironic Asserting in Transition	0	Analogy	2
Pure Challenging in Locution	13	Ironic Asserting in Locution	4	Popular Opinion	1
Pure Questioning	182	Analysing	7	Position To Know	1

Mediation **dialogue** game

Players: Px, Py and M

Locution rules:

- M can Question (PQ, AQ), Challenge (PCh) or Restate (R)
- Px and Py can assert (A), withdraw (W),
agree (Agr) or disagree (Disagr)

Commitment rules (only for Px and Py):

- After $A(p)$, performed by Px, p is added to Com-x
- After $W(p)$, performed by Px, p is removed from Com-x
- After $Agr(p)$, performed by Px, p is added to Com-x
- After $Disagr(p)$, performed by Px, p is added to Com-x

Mediation dialogue game

12 structural rules:

- (1) Px and Py can only perform one move per turn
- (2) M can perform a maximum of two moves per turn iff the first move consists of restating (R)
- (3) The dialogue starts with M seeking Px and Py's respective points of view:
 - M moves first with $PQ(t)$ addressed to Px
 - After that, Px must answer with $A(p)$
 - Then, M moves with $PQ(t)$ addressed to Py
 - Next, Py must answer with $A(q)$
- (4) The second step is to discover Px and Py's grounds for p and q , therefore:
 - M performs $PCh(p)$ addressed to Px
 - After that, Px must answer with $A(r)$
 - Then, M performs $PCh(q)$ addressed to Py
 - Next, Py must answer with $A(s)$

Mediation dialogue game

- (5) After P_x performed $A(p)$, M can perform:
 - $PQ(p)$ addressed at P_y
 - $AQ(p)$ addressed at P_y
 - $PCh(p)$ addressed at P_x
- (6) After M performed $PQ(p)$ addressed at P_x , P_x can perform:
 - $A(p)$
 - $A(\text{not-}p)$
- (7) After M performed $AQ(p)$ addressed at P_x , P_x can:
 - $W(p)$
 - $Agr(p)$
 - $Disagr(p)$
- (8) After M performed $PCh(p)$ to P_x , P_x can:
 - $A(q)$
 - $W(p)$

Mediation dialogue game

- (9) After P_x performed $W(p)$, M can:
 - $AQ(\text{not-}p)$ addressed to P_x
 - $PQ(q)$ addressed either to P_x or P_y
 - $AQ(q)$ addressed either to P_x or P_y
- (10) After P_x performed $Agr(p)$, M can:
 - $PQ(q)$ addressed either to P_x or P_y
 - $AQ(q)$ addressed either to P_x or P_y
- (11) After P_x performed $Disagr(p)$, M can:
 - $PQ(q)$ addressed to any player
 - $R(\text{not-}p)$ addressed to P_x and P_y
- (12) After M performed $R(\text{not-}p)$, M must either:
 - $AQ(\text{not-}p)$ addressed to P_x i.e. the player who previously disagreed on p , or
 - $PCh(\text{not-}p)$ addressed to P_x i.e. the player who previously disagreed on p

```

System{Mediation{
  turns{magnitude:multiple, ordering:strict}
  roles{speaker, listener, Mediator, Party}
  players{min:3, max:3}
  player{id:Mediator, roles{Mediator}}
  player{id:PartyOne, roles{Party}}
  player{id:PartyTwo, roles{Party}}

  store{id:CommitmentsPartyOne, owner:PartyOne, structure:set, visibility:public, {""}}
  store{id:CommitmentsPartyTwo, owner:PartyOne, structure:set, visibility:public, {""}}
  backtrack{on}

  /* -- Rules -- */
  rule{id:StartingRule, scope:initial,
    { assign(Mediator, speaker)
      & move(add, next, PureQuestion, PartyOne, {p}, Mediator)
    }
  }

  /* Interactions */
  interaction{PureQuestion, $Participant, {p}, PureQuestioning, {p}, "Do you believe $p?",
    { move(add, next, Assert, {p}, Target)
      & move(add, next, Assert, {!p}, Target)
    }
  }

  interaction{AssertiveQuestion, $Participant, {p}, AssertiveQuestioning, {p}, "Do you agree $p?",
    { move(add, next, Withdraw, {p}, Target)
      & move(add, next, Agree, {p}, Target)
      & move(add, next, Disagree, {p}, Target)
    }
  }

  interaction{PureChallenge, $Participant, {p}, PureChallenging, {p}, "Why $p?",
    { move(add, next, Assert, {q}, Target)
      & move(add, next, Withdraw, {p}, Target)
    }
  }

  interaction{Assert, {p}, Asserting, {p}, "I assert $p",
    {
      if{ player(PartyOne) } then
        { store(add, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, PartyTwo, {p}, Mediator)
          & move(add, next, AssertiveQuestion, PartyTwo, {p}, Mediator)
          & move(add, next, PureChallenge, PartyOne, {p}, Mediator)
        }
      else
        { store(add, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, PartyOne, {p}, Mediator)
          & move(add, next, AssertiveQuestion, PartyOne, {p}, Mediator)
          & move(add, next, PureChallenge, PartyTwo, {p}, Mediator)
          & assign(Mediator, speaker)
        }
    }
  }

  interaction{Withdraw, {p}, Asserting, {p}, "Withdraw $p",
    {
      if{ player(PartyOne) } then
        { store(remove, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, AssertiveQuestion, PartyOne, {!p}, Mediator)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
      else
        { store(remove, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, AssertiveQuestion, PartyTwo, {!p}, Mediator)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
    }
  }

  interaction{Agree, {p}, Asserting, {p}, "I agree with $p",
    {
      if{ player(PartyOne) } then
        { store(add, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, $Party, {q}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
      else
        { store(add, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, $Party, {q}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
    }
  }

  interaction{Disagree, {p}, Asserting, {!p}, Disagreeing, {<!p>,{p}>, DefaultConflict, "I disagree with $p",
    {
      if{ player(PartyOne) } then
        { store(add, {!p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, Restate, {!p}, Mediator)
        }
      else
        { store(add, {!p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, Restate, {!p}, Mediator)
        }
    }
  }

  interaction{Restate, {p}, Restating, {p}, "$p",
    { move(add, next, AssertiveQuestion, $Party, {p}, Mediator)
      & move(add, next, PureChallenge, $Party, {p}, Mediator)
    }
  }
}
}

```

```

System{Mediation{
  turns{magnitude:multiple, ordering:strict}
  roles{speaker, listener, Mediator, Party}
  players{min:3, max:3}
  player{id:Mediator, roles{Mediator}}
  player{id:PartyOne, roles{Party}}
  player{id:PartyTwo, roles{Party}}

  store{id:CommitmentsPartyOne, owner:PartyOne, structure:set, visibility:public, {""}}
  store{id:CommitmentsPartyTwo, owner:PartyOne, structure:set, visibility:public, {""}}
  backtrack{on}

  /* -- Rules -- */
  rule{id:StartingRule, scope:initial,
    { assign(Mediator, speaker)
      & move(add, next, PureQuestion, PartyOne, {p}, Mediator)
    }
  }

  /* Interactions */
  interaction{PureQuestion, $Participant, {p}, PureQuestioning, {p}, "Why $p?",
    { move(add, next, Assert, {p}, Target)
      & move(add, next, Assert, {!p}, Target)
    }
  }

  interaction{AssertiveQuestion, $Participant, {p}, Asserting, {p}, "I assert $p",
    { move(add, next, Withdraw, {p}, Target)
      & move(add, next, Agree, {p}, Target)
      & move(add, next, Disagree, {p}, Target)
    }
  }

  interaction{PureChallenge, $Participant, {p}, PureChallenging, {p}, "Why $p?",
    { move(add, next, Assert, {q}, Target)
      & move(add, next, Withdraw, {p}, Target)
    }
  }

  interaction{Assert, {p}, Asserting, {p}, "I assert $p",
    {
      if{ player(PartyOne) } then
        { store(add, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, PartyTwo, {p}, Mediator)
          & move(add, next, AssertiveQuestion, PartyTwo, {p}, Mediator)
          & move(add, next, PureChallenge, PartyOne, {p}, Mediator)
        }
      else
        { store(add, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, PartyOne, {p}, Mediator)
          & move(add, next, AssertiveQuestion, PartyOne, {p}, Mediator)
          & move(add, next, PureChallenge, PartyTwo, {p}, Mediator)
          & assign(Mediator, speaker)
        }
    }
  }

  interaction{Withdraw, {p}, Asserting, {p}, "Withdraw $p",
    {
      if{ player(PartyOne) } then
        { store(remove, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, AssertiveQuestion, PartyOne, {!p}, Mediator)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
      else
        { store(remove, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, AssertiveQuestion, PartyTwo, {!p}, Mediator)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
    }
  }

  interaction{Agree, {p}, Asserting, {p}, "I agree with $p",
    {
      if{ player(PartyOne) } then
        { store(add, {p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, $Party, {q}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
      else
        { store(add, {p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, $Party, {q}, Mediator)
          & move(add, next, AssertiveQuestion, $Party, {q}, Mediator)
        }
    }
  }

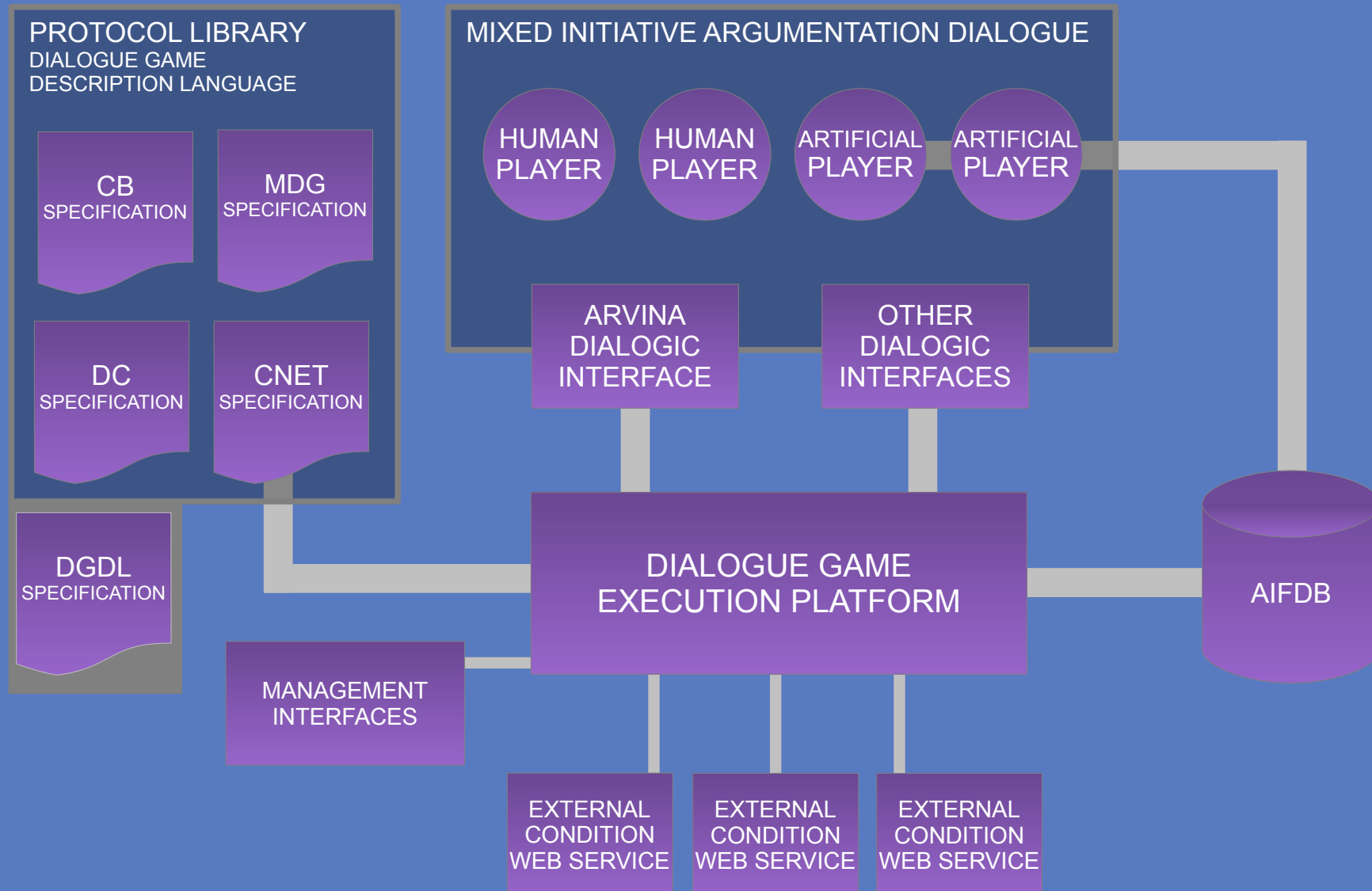
  interaction{Disagree, {p}, Asserting, {!p}, Disagreeing, {<!p>,{p}>, DefaultConflict, "I disagree with $p",
    {
      if{ player(PartyOne) } then
        { store(add, {!p}, CommitmentsPartyOne, PartyOne)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, Restate, {!p}, Mediator)
        }
      else
        { store(add, {!p}, CommitmentsPartyTwo, PartyTwo)
          & move(add, next, PureQuestion, $Party, {p}, Mediator)
          & move(add, next, Restate, {!p}, Mediator)
        }
    }
  }

  interaction{Restate, {p}, Restating, {p}, "$p",
    { move(add, next, AssertiveQuestion, $Party, {p}, Mediator)
      & move(add, next, PureChallenge, $Party, {p}, Mediator)
    }
  }
}
}

```

<http://arg.tech/MDG>

Dialogue execution




Executing MDG


The screenshot displays the Arvina Mediation Dialogue Game interface. The top left features the 'arvina' logo. Below it, a 'Participants' sidebar lists three roles: Mediator, Viv, and Eric, each with a small profile picture. Below the participants list is a 'Live Discussion Map' section, which is currently empty except for a 'Click to view' link. The main chat area on the right shows a sequence of messages:

- Mediator: "Some feeling of trust that I can do the job."
- Mediator: Viv, "Why?"
- Viv: "I have done other jobs, and I need to be able to, to be allowed to prove myself. That's it I suppose."
- Mediator: "You need to prove yourself."
- Mediator: Eric, "Do you agree?"

At the bottom of the chat area, there is a light purple box with the text: "Select a move: No moves available". The bottom of the interface has a dark blue bar with the text "arvina by ARO-tech".


Executing MDG


**Eric**


**Viv**


Case notes:


Viv has started a new job. She feels, however, that her new boss, Eric, is giving her too much to do. They have agreed to attempt mediation to try and resolve the dispute


**Eric:**
The amount of work is suitable for the hours she is contracted

**Mediator:**
Eric, why is "I've only given Viv what her contract says" the case?

**Viv:**
Yes, I have been given too much work for the hours I have

**Mediator:**
Viv, do you believe you are overworked?

**Eric:**
No, I've only given Viv what her contract says

**Mediator:**
Eric, do you believe Viv is overworked?

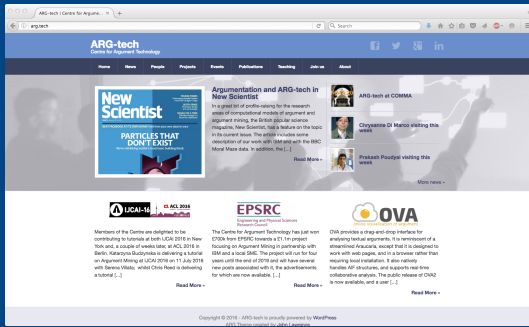
Ask:

Eric

 Why is "I have been given too much work for the hours I have"

Send

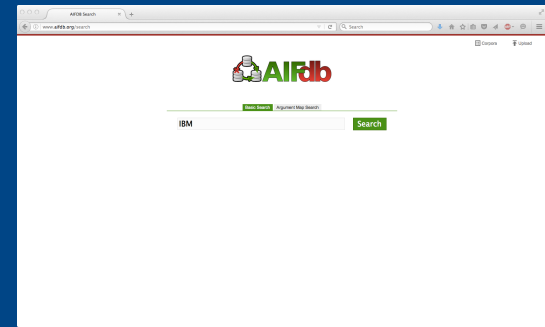
Concluding Remarks



Find out more at
<http://arg.tech>



Come to the
ESSLI 2017 Tutorial
(in Toulouse)



Investigate the
datasets at
<http://aifdb.org>

chris@arg.tech