Distributed Adaptation of Dining Philosophers

Luuk Groenewegen

Leiden University

FACS 2010
Guimarães, October 14-16, 2010

joint work with Suzana Andova and Erik de Vink

extended presentation version
Outline

- Problem situation
- Idea of solution: coordination
- Solution for unforeseen coordination
- Conclusion
Coordination problem: dynamic consistency

- **vertical** dynamic consistency
- **horizontal** dynamic consistency
UML-like collaboration diagrams during migration

Adaptation problem: dynamic consistency again
**UML-like collaboration diagrams during migration**

Adaptation problem: dynamic consistency again
- during migration: on-the-fly / no quiescence
UML-like collaboration diagrams during migration

Adaptation problem: dynamic consistency again
- during migration: on-the-fly / no quiescence
- distributed
UML-like collaboration diagrams during migration

Adaptation problem: dynamic consistency again
- during migration: on-the-fly / no quiescence
- distributed
- formal analysis of migration trajectories
coordination and adaptation

normal, foreseen coordination can be specified

- through dynamicity of temporary constraints
coordination and adaptation

normal, foreseen coordination can be specified
  through dynamicity of temporary constraints

intuitive observation:

adaptation is like unforeseen coordination
  full As-Is has starting constraints
  full To-Be has final constraints
coordination and adaptation

normal, foreseen coordination can be specified
  through dynamicity of temporary constraints

intuitive observation:

adaptation is like unforeseen coordination
  full As-Is has starting constraints
  full To-Be has final constraints
  in between are intermediate constraints
coordination and adaptation

coordination language Paradigm:

with respect to any participant of a collaboration

- constraint imposed on behaviour: phase
- constraint committed to: trap of a phase
coordination and adaptation

coordination language Paradigm:

with respect to any participant of a collaboration

- constraint imposed on behaviour: *phase*
- constraint committed to: *trap* of a phase
- the language Paradigm guarantees vertical consistency through its syntax
coordination and adaptation

migration coordination of any Paradigm model towards another Paradigm model is carried out via Paradigm pattern / component McPal
coordination and adaptation

migration coordination of any Paradigm model towards another Paradigm model is carried out via Paradigm pattern / component McPal

- JIT modeling of new adaptation as extension
- while keeping As-Is constraints
coordination and adaptation

migration coordination of any Paradigm model towards another Paradigm model is carried out via Paradigm pattern / component McPal

- JIT modeling of new adaptation as extension
- while keeping As-Is constraints
- then, migrating as extension specifies
coordination and adaptation

migration coordination of any Paradigm model towards another Paradigm model is carried out via Paradigm pattern / component *McPal*

- JIT modeling of new adaptation as extension
- while keeping As-Is constraints
- then, migrating as extension specifies
- last, shrinking from extension to To-Be
animation of Dining Philosophers

an example

5 Philosophers and 5 Forks
animation of Dining Philosophers

Paradigm model specifies coordination between

- 5 $Phil_i$ participants
animation of Dining Philosophers

Paradigm model specifies coordination between

- 5 $Phil_i$ participants
- 5 $Fork_i$ participants
- each $Fork_i$ shared between $Phil_{i-1}$ and $Phil_i$
animation of Dining Philosophers

Paradigm model coordinates
- behaviour of 5 $Phil_i$
animation of Dining Philosophers

Paradigm model coordinates

- behaviour of 5 $Phil_i$
- behaviour of 5 $Fork_i$
animation of Dining Philosophers

Paradigm model organizes Phil–Fork collaboration
animation of Dining Philosophers

Paradigm model organizes \textit{Phil–Fork} collaboration
as 5 distributed collaborations \textit{Phil2Forks};
animation of Dining Philosophers

Paradigm model organizes $Phil_i$–$Fork_i$ collaboration

- as 5 distributed collaborations $Phil2Forks_i$
- each having as participants: $Phil_i$
animation of Dining Philosophers

Paradigm model organizes $Phil_i$−$Fork_i$ collaboration
- as 5 distributed collaborations $Phil2Forks_i$
- each having as participants: $Phil_i$, $Fork_i$
**Problem situation**

**Idea of solution:** coordination

**Solution for unforeseen coordination**

**Conclusion**

---

**animation of Dining Philosophers**

Paradigm model organizes *Phil−Fork* collaboration

- as 5 distributed collaborations *Phil2Forks*;
- each having as participants: *Phil*, *Fork*, *Fork*+1;
animation of Dining Philosophers

Phil$_i$ behaviour is coordinated
animation of Dining Philosophers

Phil$_i$ behaviour is coordinated
- by constraining it
animation of Dining Philosophers

Phil\textsubscript{i} behaviour is coordinated
- by constraining it at its port
animation of Dining Philosophers

**Phil**ₖ behaviour is coordinated

- by constraining it at its port
- via imposing phases on it
**animation of Dining Philosophers**

**Phil**\textsubscript{i} behaviour is coordinated
- by constraining it at its port
- via imposing phases on it
- via committing to traps by it
animation of Dining Philosophers

Paradigm model constrains $Phil_i$ behaviour
animation of Dining Philosophers

Paradigm model constrains \textit{Phil}_i \ behaviour via

- phase \textit{Disallowed}: no eating
animation of Dining Philosophers

Paradigm model constrains $Phil_i$ behaviour via

- phase *Disallowed*: no eating
- trap *request*: wanting to eat
animation of Dining Philosophers

Paradigm model constrains $\text{Phil}_i$ behaviour via

- phase $\text{Disallowed}$: no eating
- trap $\text{request}$: wanting to eat
- phase $\text{Allowed}$: eating
**animation of Dining Philosophers**

Paradigm model constrains *Phil_i* behaviour via

- phase **Disallowed**: no eating
- trap **request**: wanting to eat
- phase **Allowed**: eating
- trap **done**: giving up eating
**Animation of Dining Philosophers**

Paradigm model gives \( \text{Phil}_i \)'s constraint dynamicity through role \( \text{Phil}_i(\text{Eater}) \).
animation of Dining Philosophers

Paradigm model gives $Phil_i$’s constraint dynamicity through role $Phil_i(Eater)$

- phases are role states
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of Dining Philosophers

Paradigm model gives $Phil_i$’s constraint dynamicity through role $Phil_i(Eater)$

- phases are role states
- connecting traps label phase transfers
Paradigm model gives $Phil_i$’s constraint dynamicity through role $Phil_i(Eater)$

- phases are role states
- connecting traps label phase transfers
animation of Dining Philosophers

wrt $\text{Phil}_i$’s left hand:
animation of Dining Philosophers

wrt $\text{Phil}_i$’s left hand: constraints on $\text{Fork}_i$

- arrive at its port-b
animation of Dining Philosophers

wrt $Phil_i$’s left hand: constraints on $Fork_i$

- arrive at its port-b
- are: phase Freed
animation of Dining Philosophers

wrt Phil_i’s left hand: constraints on Fork_i

- arrive at its port-b
- are: phase Freed ; trap gone
animation of Dining Philosophers

wrt Phil_i’s left hand: constraints on Fork_i

- arrive at its port-b
- are: phases Freed, Claimed; trap gone
animation of Dining Philosophers

wrt Phil_i’s left hand: constraints on Fork_i are

- arrive at its port-b

- are: phases Freed, Claimed; traps gone, got
animation of Dining Philosophers

wrt *Phil$_i$’s left hand*: constraints on *Fork$_i$

- arrive at its port-b
- are: phases *Freed*, *Claimed*; traps *gone*, *got*
- and via role *Fork$_i$(ForLH)*
**animation of Dining Philosophers**

wrt *Phil*$_i$’s left hand: constraints on *Fork$_i$*

- arrive at its port-b
- are: phases *Freed*, *Claimed*; traps *gone*, *got*
- and are governed via role *Fork$_i$*(*ForLH*)
*animation of Dining Philosophers*

wrt $Phil_{i-1}$'s right hand:
animation of Dining Philosophers

wrt Phil\(_{i-1}\)’s right hand: constraints on Fork\(_i\)

- arrive at its port-a
animation of Dining Philosophers

wrt $Phil_{i-1}$’s right hand: constraints on $Fork_i$
- arrive at its port-a
- are: phase Freed
wrt $\text{Phil}_{i-1}$'s right hand: constraints on $Fork_i$

- arrive at its port-

- are: phase $Freed$ ; trap $gone$
wrt $\text{Phil}_{i-1}$’s right hand: constraints on $\text{Fork}_i$

- arrive at its port-a
- are: phases Freed, Claimed; trap gone
animation of Dining Philosophers

wrt $\text{Phil}_{i-1}$’s right hand: constraints on $\text{Fork}_i$

- arrive at its port-a
- are: phases $\text{Freed}$, $\text{Claimed}$; traps $\text{gone}$, $\text{got}$
animation of Dining Philosophers

wrt $\text{Phil}_{i-1}$'s right hand: constraints on $\text{Fork}_i$

- arrive at its port-a
- are: phases $\text{Freed}$, $\text{Claimed}$; traps $\text{gone}$, $\text{got}$
- and via role $\text{Fork}_i(\text{ForRH})$
wrt $Phil_{i-1}$’s right hand: constraints on $Fork_i$

- arrive at its port-a
- are: phases $Freed$, $Claimed$; traps $gone$, $got$
- and are governed via role $Fork_i(ForRH)$
animation of Dining Philosophers

Where to start?
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
**Problem situation**

Idea of solution: coordination

Solution for unforeseen coordination

**Conclusion**

*animation of as-is coordination*
animation of as-is coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

How to move on?
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

consistency rule

* $\text{Phil}_{i-1}(\text{Eater}): \text{Disallowed}\xrightarrow{\text{request}} \text{Disallowed}$,
$\text{Fork}_{i-1}(\text{ForLH}): \text{Freed}\xrightarrow{\text{gone}} \text{Claimed}$

is now applicable
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

two consistency rules

*  \( \text{Fork}_{i-1}(\text{ForLH}) : \text{Claimed} \xrightarrow{\text{got}} \text{Claimed} \),
  \( \text{Fork}_i(\text{ForRH}) : \text{Freed} \xrightarrow{\text{gone}} \text{Claimed} \)

*  \( \text{Phil}_i(\text{Eater}) : \text{Disallowed} \xrightarrow{\text{request}} \text{Disallowed} \),
  \( \text{Fork}_i(\text{ForLH}) : \text{Freed} \xrightarrow{\text{gone}} \text{Claimed} \)

are applicable, simultaneously
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

consistency rule

\[ \ast \quad Fork_{i+1}(ForRH): Claimed \xrightarrow{\text{got}} Claimed, \]
\[ Phil_i(Eater): Disallowed \xrightarrow{\text{request}} Allowed \]

is now applicable
animation of as-is coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of as-is coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

consistency rule

* $\text{Phil}_i(Eater): \text{Allowed} \xrightarrow{\text{done}} \text{Disallowed}$,

$\text{Fork}_i(\text{ForLH}): \text{Claimed} \xrightarrow{\text{got}} \text{Freed}$,

$\text{Fork}_{i+1}(\text{ForRH}): \text{Claimed} \xrightarrow{\text{got}} \text{Freed}$

is now applicable
animation of as-is coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of as-is coordination

\[
\text{Ph}2\text{F}^-(i-1) \quad \text{Phil}^-(i-1) \quad \text{Fork}^i \quad \text{Phil}^i \quad \text{Fork}^{i+1} \quad \text{Fork}^{i-1} \quad \text{Ph}2\text{F}^i
\]
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination
animation of as-is coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of as-is coordination

consistency rule

\[ * \text{Fork}_i(\text{ForLH}): \text{Claimed} \xrightarrow{\text{got}} \text{Claimed}, \]

\[ \text{Fork}_{i+1}(\text{ForRH}): \text{Freed} \xrightarrow{\text{gone}} \text{Claimed} \]

is now applicable
animation of as-is coordination
animation of migration coordination

... and all the time

McPal is in hibernation
animation of migration coordination
**animation of migration coordination**

---

Suzana Andova & Luuk Groenewegen & Erik de Vink

Distributed Adaptation of Dining Philosophers
animation of migration coordination

though hibernating,

McPal prepares an adaptation
animation of migration coordination
animation of migration coordination

and $McPal$ decides to achieve for to-be situation again

5 $Phil2Forks_i$ collaborations

but not LeftFirst for each
animation of migration coordination

of these 5 collaborations

- LeftFirst for 3
- RightFirst for 2
animation of migration coordination

of these 5 collaborations

- LeftFirst for 3 or for 2
- RightFirst for 2 or for 3
animation of migration coordination

of these 5 collaborations

- LeftFirst for 3 or for 2
- RightFirst for 2 or for 3

- the 2 or 2 aren’t neighbours
animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of migration coordination

once McPal has clear what to do and how, McPal extends the model while keeping all constraints
animation of migration coordination

via consistency rule

\[ \text{McPal} : \text{JITting}^{\text{giveOut}} \rightarrow \text{StartMigr} * \]

\[ \text{McPal} : [\text{Crs} := \text{Crs} + \text{Crs}_{\text{migr}} + \text{Crs}_{\text{toBe}}] \]
animation of migration coordination
animation of migration coordination
animation of migration coordination

so, more description is in place but dynamics remains as-is
animation of migration coordination
animation of migration coordination

and more description is in place

while dynamics remains as-is
animation of migration coordination
animation of migration coordination
animation of migration coordination

via choreography rule

\[ \star \text{McPal(Evol)} : \text{Hibernating} \xrightarrow{\text{prepared}} \text{Migrating} \]

first

\textit{McPal}'s own migration begins
animation of migration coordination
animation of migration coordination
animation of migration coordination
animation of migration coordination

orchestration rule

\[ \text{McPal} : \text{StartMigr} \xrightarrow{\text{create}} \text{Delegated} \]

\[
\begin{align*}
\text{McPhil}_1(\text{Evol}) & : \text{Passive} \xrightarrow{\text{triv}} \text{Active}, \\
\text{McPhil}_2(\text{Evol}) & : \text{Passive} \xrightarrow{\text{triv}} \text{Active}, \ldots \\
\ldots, \text{McPhil}_5(\text{Evol}) & : \text{Passive} \xrightarrow{\text{triv}} \text{Active}, \\
\text{McPal} & : [\text{Crs} := \text{Crs}_{\text{noHb}} + \text{Crs}_{\text{hibr}}] \\
\text{McPhil}_1 & : [\text{Crs}_1 := \text{Crs}_{1,\text{asIs}} + \text{Crs}_{1,\text{orch}}], \ldots \\
\ldots, \text{McPhil}_5 & : [\text{Crs}_5 := \text{Crs}_{5,\text{asIs}} + \text{Crs}_{5,\text{orch}}] \\
\text{is now applicable}
\]
animation of migration coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination
animation of migration coordination

orchestration rule

\[ McPhil_{i-1} : \text{Awake} \xrightarrow{\text{takeOver}} \text{JoiningIn} \ast \]

\[ McPhil_{i-1} : [Crs_{i-1} := Crs_{i-1} - Crs_{i-1,asls}] \]

is now applicable, keeping an orchestrated equivalent of ongoing as-is choreography
animation of migration coordination
animation of migration coordination
animation of migration coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination

orchestration rule

\[ McPhil_i : \text{Awake} \xrightarrow{\text{takeOver}} \text{JoiningIn} \ast \]

\[ McPhil_i : [Crs_i := Crs_i - Crs_i,_{asls}] \]

is now applicable
animation of migration coordination
animation of migration coordination

and orchestration rule

$McPh{i-1} : JoiningIn \xrightarrow{\text{conductToR}} ToR \ast \ldots$

$\ldots \quad 1 \text{ out of 3 cases}$

is now applicable too
animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination

orchestration rule

\[ \text{McPhil}_i : \text{JoiningIn} \xrightarrow{\text{conductToR}} \text{ToR} \ast \ldots \]

\[ \ldots \quad 1 \text{ out of 3 cases} \]

is now applicable
animation of migration coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion
animation of migration coordination
**Problem situation**

**Idea of solution:** coordination

**Solution for unforeseen coordination**

**Conclusion**

**animation of migration coordination**

Distributed Adaptation of Dining Philosophers
animation of migration coordination

eventually, orchestration rule

\[ \text{McPal} : \text{Delegated} \xrightarrow{\text{singleSwapRL}} \text{Gathering} \]

\[ \text{McPhil}_j(Evol) : \text{Active} \xrightarrow{\text{halfwayR}} \text{EndAsR}, \]
\[ \text{McPhil}_{j+1}(Evol) : \text{Active} \xrightarrow{\text{halfwayR}} \text{EndAsL}, \]
\[ \text{McPhil}_{j+2}(Evol) : \text{Active} \xrightarrow{\text{halfwayL}} \text{EndAsR}, \]
\[ \text{McPhil}_{j+3}(Evol) : \text{Active} \xrightarrow{\text{halfwayL}} \text{EndAsL}, \]
\[ \text{McPhil}_{j+4}(Evol) : \text{Active} \xrightarrow{\text{halfwayL}} \text{EndAsL} \]

is applicable \((\text{for } j = i - 1)\)
animation of migration coordination
animation of migration coordination
animation of migration coordination

eventually orchestration rule

\[ \text{McPhil}_i : \text{ToR}^{\text{choreofyToL}} \rightarrow \text{ToBeL} \quad * \ldots \quad * \ldots \quad 1 \text{ out of } 5 \text{ cases} \]

is applicable

keeping

a choreographical equivalent of ongoing as-is orchestration
Animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink

Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination
animation of migration coordination

orchestration rule

\[\text{McPal} : \text{Gathering}^{\text{collect}} \rightarrow \text{Gathering} \ast\]

\[\text{McPhil}_i(Evol) : \text{EndAsL}^{\text{done}} \rightarrow \text{Retreating},\]

\[\text{McPal} : [\text{Crs} := \text{Crs} + \text{Crs}_{i,\text{toBe}}, \]

\[\text{Crs}_{\text{toBe}} := \text{Crs}_{\text{toBe}} + \text{Crs}_{i,\text{toBe}}]\]

is now applicable
animation of migration coordination
animation of migration coordination
animation of migration coordination

eventually orchestration rule

\[ \text{McPhil}_{i-1} : \text{ToR}^{\text{choreofyToR}} \rightarrow \text{ToBeR} \ast \ldots \]

\[ \ldots 1 \text{ out of 4 cases} \]

is applicable
keeping

a choreographical equivalent of ongoing as-is orchestration.
animation of migration coordination
animation of migration coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion
animation of migration coordination
animation of migration coordination

orchestration rule

\[ McPal : Gathering^{collect} \rightarrow Gathering \ast \]

\[ McPhil_{i-1}(Evol) : EndAsL^{done} \rightarrow Retreating, \]

\[ McPal : [Crs := Crs + Crs_{i-1,toBe}, \]

\[ Crs_{toBe} := Crs_{toBe} + Crs_{i-1,toBe}] \]

is now applicable
animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink

Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion
animation of migration coordination
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

**animation of migration coordination**

Suzana Andova & Luuk Groenewegen & Erik de Vink

Distributed Adaptation of Dining Philosophers
animation of migration coordination

eventually, orchestration rule

McPal : Gathering $\xrightarrow{\text{close}}$ Content $\ast$

$\text{McPhil}_1(\text{Evol}) : \text{Retreating} \xrightarrow{\text{away}} \text{Passive}$,
$\text{McPhil}_2(\text{Evol}) : \text{Retreating} \xrightarrow{\text{away}} \text{Passive}$,
$\text{McPhil}_3(\text{Evol}) : \text{Retreating} \xrightarrow{\text{away}} \text{Passive}$,
$\text{McPhil}_4(\text{Evol}) : \text{Retreating} \xrightarrow{\text{away}} \text{Passive}$,
$\text{McPhil}_5(\text{Evol}) : \text{Retreating} \xrightarrow{\text{away}} \text{Passive}$

is applicable
animation of migration coordination
animation of migration coordination
animation of migration coordination
animation of as-is coordination

consistency rule

* $\text{McPal}(\text{Eater})$: Migrating $\xrightarrow{\text{done}}$ Hibernating

is now applicable
Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion
animation of migration coordination

Problem situation
Idea of solution: coordination
Solution for unforeseen coordination
Conclusion

Suzana Andova & Luuk Groenewegen & Erik de Vink
Distributed Adaptation of Dining Philosophers
animation of migration coordination
animation of migration coordination

via consistency rule

$McPal : \text{Content}^{\text{cleanUp}} \rightarrow \text{Observing} \ast$

$McPal : [\text{Crs} := \text{Crs}_{\text{toBe}}]$

$McPal$ shrinks the model
animation of migration coordination

Suzana Andova & Luuk Groenewegen & Erik de Vink

Distributed Adaptation of Dining Philosophers
animation of migration coordination

... and so on and so forth ...
concluding remarks

- distributed self-adaptation
- on-the-fly, without any quiescence
- creation / deletion of components
- computation of behaviour
- zipless swapping choreography – orchestration
future directions

- model checking with mCRL2 / Prism
- larger examples
- applications: Edafmis, network maintenance
- migration patterns