1 Lecture 1: Introduction

2 Lecture 2: Message Sequence Charts

<ロト <問ト < 国ト < 国ト

RWTH

Theoretical Foundations of the UML Lecture 1: Introduction

Joost-Pieter Katoen

Lehrstuhl für Informatik 2 Software Modeling and Verification Group

moves.rwth-aachen.de/teaching/ss-16/theoretical-foundations-of-the-uml/

13. April 2016

Joost-Pieter Katoen Theoretical Foundations of the UML

(4) E (4) E

< 口 > < 同 >

Target audience

You are studying:

- Master Computer Science, or
- Master Systems Software Engineering, or
- Bachelor Computer Science, or
-



Target audience

You are studying:

- Master Computer Science, or
- Master Systems Software Engineering, or
- Bachelor Computer Science, or

•

Usage as:

- elective course Theoretical Computer Science
- not a Wahlpflicht course for bachelor students
- specialization MOVES (Modeling and Verification of Software)
- complementary to Model-based Software Development (Rumpe)

(4) (5) (4) (5)

In general:

- interest in system software engineering
- interest in formal methods for software
- interest in semantics and verification
- application of mathematical reasoning

3 x 4 3

In general:

- interest in system software engineering
- interest in formal methods for software
- interest in semantics and verification
- application of mathematical reasoning

Prerequisites:

- mathematical logic
- formal language and automata theory
- algorithms and data structures
- computability and complexity theory

Schedule:						
	Day	Time	Room			
Lecture	Tue	12:15 - 13:45	9U09			
	Thu	08:30 - 10:15	5056			
Exercises	Thu	14:15- 15:45	9U09			
about 20 lectures in total; Keep track of website for precise dates!						



<ロト <問ト < 注ト < 注ト

RWTHAACHEN UNIVERSITY

æ

	Day	Time	Room		
Lecture	Tue	12:15 - 13:45	9U09		
	Thu	08:30 - 10:15	5056		
Exercises	Thu	14:15- 15:45	9U09		
about 20 lectures in total; Keep track of website for precise dates!					

People involved:					
	Lecturer	EMail			
Lectures	Joost-Pieter Katoen	katoen@cs.rwth-aachen.de			
Exercises	Hao Wu	hao.wu@cs.rwth-aachen.de			
	Benjamin Kaminksi	benjamin.kaminski@cs.rwth-aachen.de			

Joost-Pieter Katoen Theoretical Foundations of the UML

◆ロト ◆御ト ◆注ト ◆注ト ─注

Assignments:

- (almost) weekly assignments
- available from course web-site
- first assignment: Thursday April 21
- hand in solution at start next exercise class

Joost-Pieter Katoen

- groups of maximally two students
- first exercise class: Thursday April 28

Examination: (6 ECTS credit points)

- written exam: July XY, 2016 (tba)
- written re-exam: September WZ, 2016 (tba)



(신문) (문)

< 同 >

Examination: (6 ECTS credit points)

- written exam: July XY, 2016 (tba)
- written re-exam: September WZ, 2016 (tba)

Admission:

• at least 40% of exercise points



Joost-Pieter Katoen Theoretical Foundations of the UML

Motivation

Scope:

- Goal: formal description + analysis of (concurr.) software systems
- Focus: the <u>Unified Modeling Language</u>



~ 프 > ~ 프 >

< 同 ▶

Motivation

Scope:

- Goal: formal description + analysis of (concurr.) software systems
- Focus: the <u>Unified Modeling Language</u>

More specifically:

- Sequence Diagrams (used for requirements analysis)
- Propositional Dynamic Logic
- Communicating Finite State Automata
- Statecharts (behavioral description of systems)

Motivation

Scope:

- $\bullet\,$ Goal: formal description + analysis of (concurr.) software systems
- Focus: the <u>Unified Modeling Language</u>

More specifically:

- Sequence Diagrams (used for requirements analysis)
- Propositional Dynamic Logic
- Communicating Finite State Automata
- Statecharts (behavioral description of systems)

Aims:

- clarify and make precise the semantics of some UML fragments
- formal reasoning about basic properties of UML models
- convince you that UML models are much harder than you think

What is it ******not****** about?

- the use of the UML in the software development cycle
 - see the complementary course by Prof. Rumpe
- other notations of the UML (e.g., class diagrams, activity diagrams)
- what is precisely in the UML, and what is not
 - liberal interpretation of which constructs belong to the UML
- applying the UML to concrete SW development case studies
- empirical results on the usage of UML
- drawing pictures

Ο...

→ Ξ → → Ξ →



2 Lecture 2: Message Sequence Charts



< 47 ▶

★ E ► < E ►</p>

10/32

RWTH

Theoretical Foundations of the UML Lecture 2: Message Sequence Charts

Joost-Pieter Katoen

Lehrstuhl für Informatik 2 Software Modeling and Verification Group

moves.rwth-aachen.de/teaching/ss-16/theoretical-foundations-of-the-uml/

13. April 2016

Joost-Pieter Katoen Theoretical Foundations of the UML

43643

11/32

4 - □
 ▶

- 70s 80s: often used informally
- $\bullet~1992:$ first version of MSCs standardized by CCITT (currently ITU) Z.120
- 1992 1996: many extensions, e.g., high-level + formal semantics (using process algebras)
- 1996: MSC'96 standard
- 2000: MSC 2000, time, data, o-o features
- 2005: MSC 2004 ...

프 문 문 프 문

Variants of MSCs

- UML sequence diagrams
- (instantiations of) use cases
- triggered MSCs
- netcharts (= Petri net + MSC)
- STAIRS
- Live sequence charts
- . . .

() < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < () < ()

- scenario-based language
- visual representation
- "easy" to comprehend
- generalization possible towards automata (states are MSCs)
- widely used in industrial practice

< ∃⇒

- requirements specification (positive, negative scenarios, e.g., CREWS)
- system design and software engineering
- visualization of test cases (graphical extension to TTCN)
- feature interaction detection
- workflow management systems

• . . .

Joost-Pieter Katoen Theoretical Foundations of the UML

(B)



Joost-Pieter Katoen Theoretical Foundations of the UML

16/32

RWTHAACHEN UNIVERSITY

æ

(신문) (신문)

< □ > < □ >



These pictures are formalized using partial orders.

Joost-Pieter Katoen Theoretical Foundations of the UML

E> < E>

Partial orders

Definition

Let E be a set of events.

A partial order over E is a relation $\leq E \times E$ such that:

 $\textcircled{0} \preceq \text{ is reflexive, i.e., } \forall e \in E. e \preceq e,$

- ② \leq is transitive, i.e., $e \leq e' \land e' \leq e''$ implies $e \leq e''$, and
- **③** \leq is anti-symmetric, i.e., $\forall e, e'. (e \leq e' \land e' \leq e) \Rightarrow e = e'.$

The pair (E, \preceq) is called a **partially ordered set** (poset, for short).

伺下 イヨト イヨト

Partial orders

Definition

Let E be a set of events.

A partial order over E is a relation $\leq E \times E$ such that:

- ② \leq is transitive, i.e., $e \leq e' \land e' \leq e''$ implies $e \leq e''$, and
- **③** \leq is anti-symmetric, i.e., $\forall e, e'. (e \leq e' \land e' \leq e) \Rightarrow e = e'.$

The pair (E, \preceq) is called a partially ordered set (poset, for short).

Definition

Let (E, \preceq) be a poset and let $e, e' \in E$. e and e' are comparable if $e \preceq e'$ or $e' \preceq e$. Otherwise, they are incomparable.

・ 「 ・ ・ ・ ・ ・ ・ ・ ・

Partial orders

Definition

Let E be a set of events.

A partial order over E is a relation $\leq E \times E$ such that:

- ② \leq is transitive, i.e., $e \leq e' \land e' \leq e''$ implies $e \leq e''$, and
- **③** \leq is anti-symmetric, i.e., $\forall e, e'. (e \leq e' \land e' \leq e) \Rightarrow e = e'.$

The pair (E, \preceq) is called a **partially ordered set** (poset, for short).

Definition

Let (E, \preceq) be a poset and let $e, e' \in E$. e and e' are comparable if $e \preceq e'$ or $e' \preceq e$. Otherwise, they are incomparable.

 \leq is a non-strict partial order as it is reflexive. A strict partial order is a relation \prec that is irreflexive, transitive and asymmetric (i.e., if $e \prec e'$ then not $e' \prec e$).

Let (E, \preceq) be a poset. The Hasse diagram (E, \lessdot) of (E, \preceq) is defined by:

$$e \lessdot e' \text{ iff } e \preceq e' \text{ and } \neg (\exists e'' \neq e, e'. e \preceq e'' \land e'' \preceq e')$$

Hasse diagrams can be used to visualize posets with finitely many elements in a succinct way.

Let (E, \preceq) be a poset. A linearization of (E, \preceq) is a total order $\sqsubseteq \subseteq E \times E$ such that $e \preceq e'$ implies $e \sqsubseteq e'$

A linearization is a topological sort of the Hasse diagram of (E, \preceq) . Note that every partial order has at least one linearization.

Example

Example

Let $E = \{e_1, \dots, e_6\},\$

$$\leq = \{ (e_1, e_2), (e_1, e_3), (e_3, e_4), (e_4, e_5), (e_5, e_6), (e_1, e_4), \\ (e_3, e_5), (e_1, e_5), (e_1, e_6), (e_3, e_6), (e_4, e_6) \\ \}^r \text{ where } R^r \text{ denotes the reflexive closure of } R$$

Linearizations:

- $e_1 e_2 e_3 e_4 e_5 e_6$,
- $e_1e_3e_2e_4e_5e_6$,
- $e_1e_3e_4e_2e_5e_6$,
- $e_1e_3e_4e_5e_2e_6$,
- $e_1e_3e_4e_5e_6e_2$
- No linearizations:
 - $e_2 e_1 e_3 \dots$, and $e_1 e_4 e_3 \dots$

イロト イポト イヨト イヨト

Hasse diagram:



Joost-Pieter Katoen Theoretical Foundations of the UML

э

- Let \mathcal{P} : finite set of (sequential) processes
 - \mathcal{C} : finite set of message contents $(a, b, c, \ldots \in \mathcal{C})$



- Let \mathcal{P} : finite set of (sequential) processes
 - C: finite set of message contents $(a, b, c, \ldots \in C)$

Definition

Communication action: $p, q \in \mathcal{P}, p \neq q, a \in \mathcal{C}$

- !(p,q,a) "process p sends message a to process q"
- ?(p,q,a) "process p receives message a sent by process q"

Let Act denote the set of communication actions

(人間) トイヨト イヨト

Definition

An MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ with:

Joost-Pieter Katoen Theoretical Foundations of the UML

<ロト <問ト < 注ト < 注ト

OTH VELOCIT

э

Definition

An MSC $M = (\mathcal{P}, \mathcal{E}, \mathcal{C}, l, m, \preceq)$ with:

• \mathcal{P} , a finite set of processes $\{p_1, p_2, \ldots, p_n\}$ with n > 1

Definition

An MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ with:

- \mathcal{P} , a finite set of processes $\{p_1, p_2, \ldots, p_n\}$ with n > 1
- E, a finite set of events

$$E = \biguplus_{p \in \mathcal{P}} E_p = E_? \cup E_!$$

Definition

An MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ with:

- \mathcal{P} , a finite set of processes $\{p_1, p_2, \ldots, p_n\}$ with n > 1
- E, a finite set of events

$$E = \biguplus_{p \in \mathcal{P}} E_p = E_? \cup E_!$$

• \mathcal{C} , a finite set of message contents

Definition

An MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ with:

- \mathcal{P} , a finite set of processes $\{p_1, p_2, \ldots, p_n\}$ with n > 1
- E, a finite set of events

$$E = \biguplus_{p \in \mathcal{P}} E_p = E_? \cup E_!$$

- \mathcal{C} , a finite set of message contents
- $l: E \to Act$, a labelling function defined by:

$$l(e) = \begin{cases} !(p,q,a) & \text{if } e \in E_p \cap E_! \\ ?(p,q,a) & \text{if } e \in E_p \cap E_? \end{cases}, \text{ for } p \neq q \in \mathcal{P}, a \in \mathcal{C} \end{cases}$$



Joost-Pieter Katoen Theoretical Foundations of the UML

< 177 ▶

< 注 > < 注 >

23/32

Definition

• $m: E_! \to E_?$ a bijection ("matching function"), satisfying:

 $m(e) = e' \wedge l(e) = !(p,q,a)$ implies $l(e') = ?(q,p,a) \ (p \neq q, a \in \mathcal{C})$

UNIVERSIT

Definition

• $m: E_1 \to E_?$ a bijection ("matching function"), satisfying:

$$m(e) = e' \wedge l(e) = !(p,q,a) \text{ implies } l(e') = ?(q,p,a) \ (p \neq q, \ a \in \mathcal{C})$$

• $\leq \subseteq E \times E$ is a partial order ("visual order") defined by:



where for relation R, R^* denotes its reflexive and transitive closure.

Example (1)



$$\begin{split} M &= (\mathcal{P}, E, \mathcal{C}, l, m, \preceq) \text{ with:} \\ \mathcal{P} &= \{p_1, p_2\} \qquad E_{p_1} = \{e_1, e_4\} \\ E &= \{e_1, e_2, e_3, e_4\} \qquad E_{p_2} = \{e_2, e_3\} \\ \mathcal{C} &= \{a, b\} \qquad E_! = \{e_1, e_3\}, \\ E_? &= \{e_2, e_4\} \\ l(e_1) &= !(p_1, p_2, a) \qquad m(e_1) = e_2 \\ l(e_2) &= ?(p_2, p_1, a) \\ l(e_3) &= !(p_2, p_1, b) \qquad m(e_3) = e_4 \\ l(e_4) &= ?(p_1, p_2, b) \end{split}$$

Ordering at processes: $e_1 <_{p_1} e_4$ and $e_2 <_{p_2} e_3$ Hasse diagram of (E, \preceq) : $e_1 \longrightarrow e_2 \longrightarrow e_3 \longrightarrow e_4$

Linearizations?

Joost-Pieter Katoen Theoretical Foundations of the UML

∃ ► < ∃ ►</p>

Example (2)



$$M' = (\underbrace{\mathcal{P}, E, \mathcal{C}, l, m}_{\text{as above}}, \preceq')$$
 with:





Joost-Pieter Katoen Theoretical Foundations of the UML

RWTHA

æ

(本語)) (本語))

< A

This is not an MSC



Joost-Pieter Katoen Theoretical Foundations of the UML

< □ > < □ >

RWTHAACHEN UNIVERSITY

æ

26/32

→ E → < E →</p>

FIFO property

MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ has the *First-In-First-Out* (FIFO) property whenever: for all $e, e' \in E_1$ we have

 $e \prec e' \land l(e) = !(p,q,a) \land l(e') = !(p,q,b) \text{ implies } m(e) \prec m(e')$

i.e., "no message overtaking allowed"



・ 同 ト ・ ヨ ト ・ ヨ ト

FIFO property

MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ has the *First-In-First-Out* (FIFO) property whenever: for all $e, e' \in E_1$ we have

 $e \prec e' \land l(e) = !(p,q,a) \land l(e') = !(p,q,b)$ implies $m(e) \prec m(e')$

i.e., "no message overtaking allowed"



Theoretical Foundations of the UML Joost-Pieter Katoen

FIFO property

MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, \preceq)$ has the *First-In-First-Out* (FIFO) property whenever: for all $e, e' \in E_!$ we have

 $e \prec e' \land l(e) = !(p,q,a) \land l(e') = !(p,q,b) \text{ implies } m(e) \prec m(e')$

i.e., "no message overtaking allowed"



Let Lin(M) = denote the set of linearizations of MSC M.



Let Lin(M) = denote the set of linearizations of MSC M.

MSCs and its linearizations are interchangeable

There is a one-to-one correspondence between an MSC and its set of linearizations.



(신문) (문)

Let Lin(M) = denote the set of linearizations of MSC M.

MSCs and its linearizations are interchangeable

There is a one-to-one correspondence between an MSC and its set of linearizations.

Thus:

Lin(M) uniquely characterizes the MSC M.



Let Lin(M) = denote the set of linearizations of MSC M.

MSCs and its linearizations are interchangeable

There is a one-to-one correspondence between an MSC and its set of linearizations.

Thus:

Lin(M) uniquely characterizes the MSC M.

From MSCs to its set of linearizations is straightforward. The reverse direction is discussed in the following. First: well-formedness.

イロト 不得下 イヨト イヨト

Let $Ch := \{(p,q) \mid p \neq q, p, q \in \mathcal{P}\}$ be the set of channels over \mathcal{P} .



Joost-Pieter Katoen Theoretical Foundations of the UML

29/32

Let $Ch := \{(p,q) \mid p \neq q, p, q \in \mathcal{P}\}$ be the set of channels over \mathcal{P} . We call $w = a_1 \dots a_n \in Act^*$ proper if



Let $Ch := \{(p,q) \mid p \neq q, p, q \in \mathcal{P}\}$ be the set of channels over \mathcal{P} .

We call $w = a_1 \dots a_n \in Act^*$ proper if

• every receive in w is preceded by a corresponding send, i.e.: $\forall (p,q) \in Ch$ and prefix u of w, we have:



where $|u|_a$ denotes the number of occurrences of action a in u

・ 戸 ト ・ ヨ ト ・ ヨ ト

1

Let $Ch := \{(p,q) \mid p \neq q, p, q \in \mathcal{P}\}$ be the set of channels over \mathcal{P} .

We call $w = a_1 \dots a_n \in Act^*$ proper if

• every receive in w is preceded by a corresponding send, i.e.: $\forall (p,q) \in Ch$ and prefix u of w, we have:



where $|u|_a$ denotes the number of occurrences of action a in uthe FIFO policy is respected, i.e.: $\forall 1 \leq i < j \leq n, (p,q) \in Ch$, and $a_i = !(p,q,m_1), a_j = ?(q,p,m_2)$:

$$\sum_{n \in \mathcal{C}} |a_1 \dots a_{i-1}|_{!(p,q,m)} = \sum_{m \in \mathcal{C}} |a_1 \dots a_{j-1}|_{?(q,p,m)} \quad \text{implies} \quad m_1 = m_2$$

・ 同 ト ・ ヨ ト ・ ヨ ト

Let $Ch := \{(p,q) \mid p \neq q, p, q \in \mathcal{P}\}$ be the set of channels over \mathcal{P} .

We call $w = a_1 \dots a_n \in Act^*$ proper if

• every receive in w is preceded by a corresponding send, i.e.: $\forall (p,q) \in Ch$ and prefix u of w, we have:



where $|u|_a$ denotes the number of occurrences of action a in u2 the FIFO policy is respected, i.e.: $\forall 1 \leq i < j \leq n, (p,q) \in Ch$, and $a_i = !(p,q,m_1), a_j = ?(q,p,m_2)$: $\sum_{n \in C} |a_1 \dots a_{i-1}|_{!(p,q,m)} = \sum_{m \in C} |a_1 \dots a_{j-1}|_{?(q,p,m)}$ implies $m_1 = m_2$

A proper word w is well-formed if $\sum_{m \in \mathcal{C}} |w|_{!(p,q,m)} = \sum_{m \in \mathcal{C}} |w|_{?(q,p,m)_{O,Q,Q}}$

Proposition

For every MSC M and every $w \in Lin(M)$, w is well-formed.

Lin(M) denotes a set of words (and not linearizations) the word of linearization $e_1 \dots e_n$ equals $\ell(e_1) \dots \ell(e_n)$



Joost-Pieter Katoen Theoretical Foundations of the UML

< 177 ▶

31/32

Associate to $w = a_1 \dots a_n \in Act^*$ an Act-labelled poset

 $M(w) = (E, \preceq, \ell)$



Associate to $w = a_1 \dots a_n \in Act^*$ an Act-labelled poset

$$M(w) = (E, \preceq, \ell)$$

such that:

• $E = \{1, ..., n\}$ are the positions in w labelled with $\ell(i) = a_i$



・ 同 ト ・ ヨ ト ・ ヨ ト

Associate to $w = a_1 \dots a_n \in Act^*$ an Act-labelled poset

$$M(w) = (E, \preceq, \ell)$$

such that:

E = {1,...,n} are the positions in w labelled with l(i) = a_i
≤ = (U_{p∈P} ≺_p ∪ ≺_{msg})^{*} where
i ≺_p j if and only if i < j for every i, j ∈ E_p



・ 「 ・ ・ ・ ・ ・ ・ ・ ・

31/32

Associate to $w = a_1 \dots a_n \in Act^*$ an Act-labelled poset

$$M(w) = (E, \preceq, \ell)$$

such that:

• $E = \{1, \ldots, n\}$ are the positions in w labelled with $\ell(i) = a_i$ • $\preceq = \left(\bigcup_{p \in \mathcal{P}} \prec_p \cup \prec_{msg}\right)^*$ where • $i \prec_p j$ if and only if i < j for every $i, j \in E_p$ • $i \prec_{msg} j$ if for some $(p, q) \in Ch$ and $m \in \mathcal{C}$ we have: $\ell(i) = !(p, q, m) \text{ and } \ell(j) = ?(q, p, m) \text{ and}$ $\sum_{m \in \mathcal{C}} |a_1 \ldots a_{i-1}|_{!(p,q,m)} = \sum_{m \in \mathcal{C}} |a_1 \ldots a_{j-1}|_{?(q,p,m)}$

イロト 不得下 イヨト イヨト

Associate to $w = a_1 \dots a_n \in Act^*$ an Act-labelled poset

$$M(w) = (E, \preceq, \ell)$$

such that:

• $E = \{1, \ldots, n\}$ are the positions in w labelled with $\ell(i) = a_i$ • $\preceq = \left(\bigcup_{p \in \mathcal{P}} \prec_p \cup \prec_{msg}\right)^*$ where • $i \prec_p j$ if and only if i < j for every $i, j \in E_p$ • $i \prec_{msg} j$ if for some $(p, q) \in Ch$ and $m \in \mathcal{C}$ we have: $\ell(i) = !(p, q, m) \text{ and } \ell(j) = ?(q, p, m) \text{ and}$ $\sum_{m \in \mathcal{C}} |a_1 \ldots a_{i-1}|_{!(p,q,m)} = \sum_{m \in \mathcal{C}} |a_1 \ldots a_{j-1}|_{?(q,p,m)}$

イロト 不得下 イヨト イヨト

Associate to $w = a_1 \dots a_n \in Act^*$ an <u>Act-labelled</u> poset

$$M(w) = (E, \preceq, \ell)$$

such that:

•
$$E = \{1, ..., n\}$$
 are the positions in w labelled with $\ell(i) = a_i$
• $\preceq = \left(\bigcup_{p \in \mathcal{P}} \prec_p \cup \prec_{msg}\right)^*$ where
• $i \prec_p j$ if and only if $i < j$ for every $i, j \in E_p$
• $i \prec_{msg} j$ if for some $(p,q) \in Ch$ and $m \in \mathcal{C}$ we have:
 $\ell(i) = !(p,q,m)$ and $\ell(j) = ?(q,p,m)$ and
 $\sum_{m \in \mathcal{C}} |a_1 \dots a_{i-1}|_{!(p,q,m)} = \sum_{m \in \mathcal{C}} |a_1 \dots a_{j-1}|_{?(q,p,m)}$

Example

construct M(w) for $w = !(r,q,m)!(p,q,m_1)!(p,q,m_2)?(q,p,m_1)?(q,p,m_2)?(q,r,m)$

Relating well-formed words to MSCs

For every well-formed $w \in Act^*$, M(w) is an MSC.

Definition

 (E, \leq, ℓ) and (E', \leq', ℓ') are isomorphic if there exists a bijection $f: E \to E'$ such that $e \leq e'$ iff $f(e) \leq' f(e')$ and $\ell(e) = \ell'(f(e))$.

Linearizations yield isomorphic MSCs

For every well-formed $w \in Act^*$ and $w' \in Lin(M(w))$:

M(w) and M(w') are isomorphic.

・ロト ・ 雪 ト ・ ヨ ト