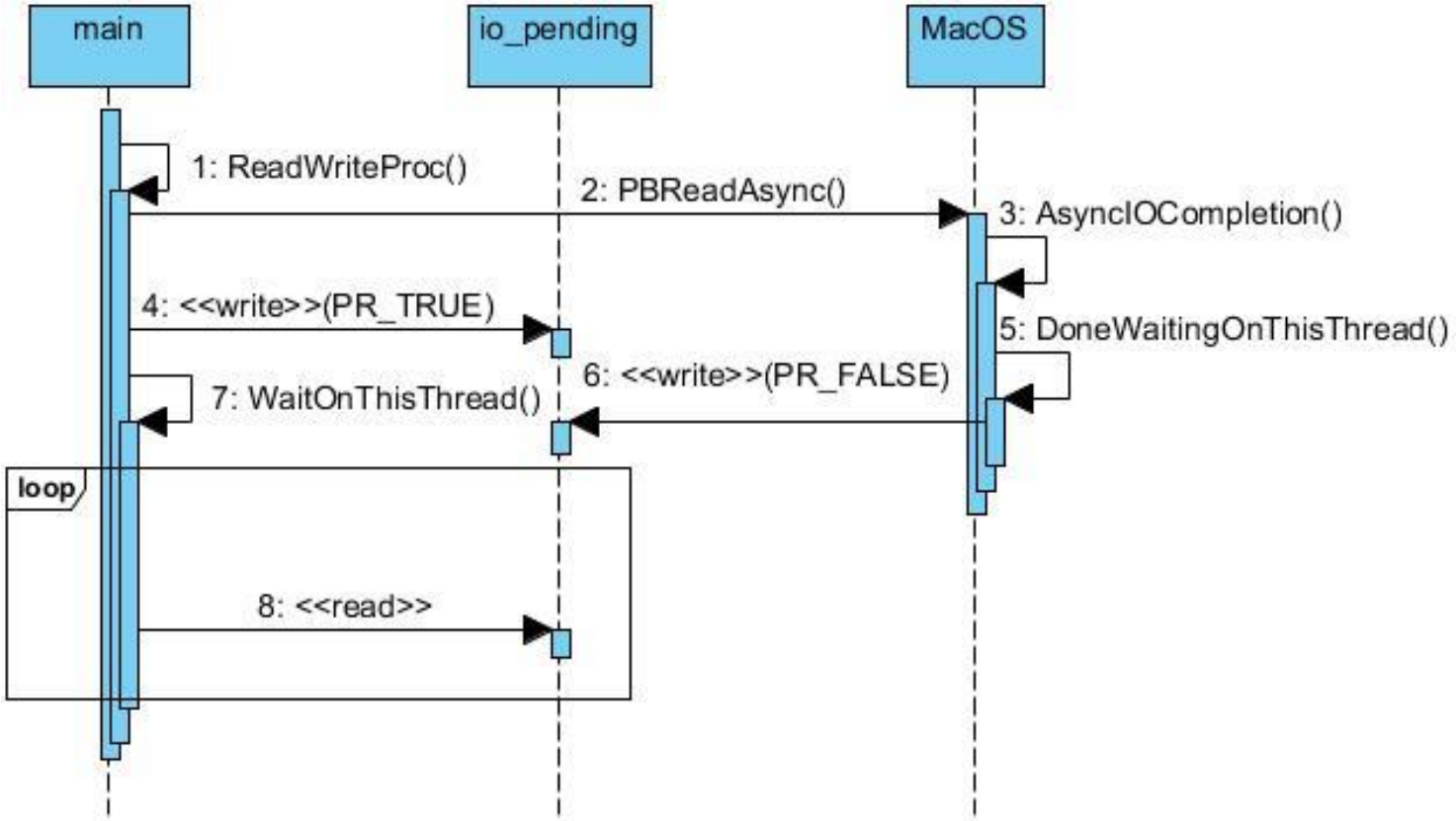
On Detecting Concurrency Defects Automatically at the Design Level

Frank Padberg
and Luis M. Carril
KIT, Germany

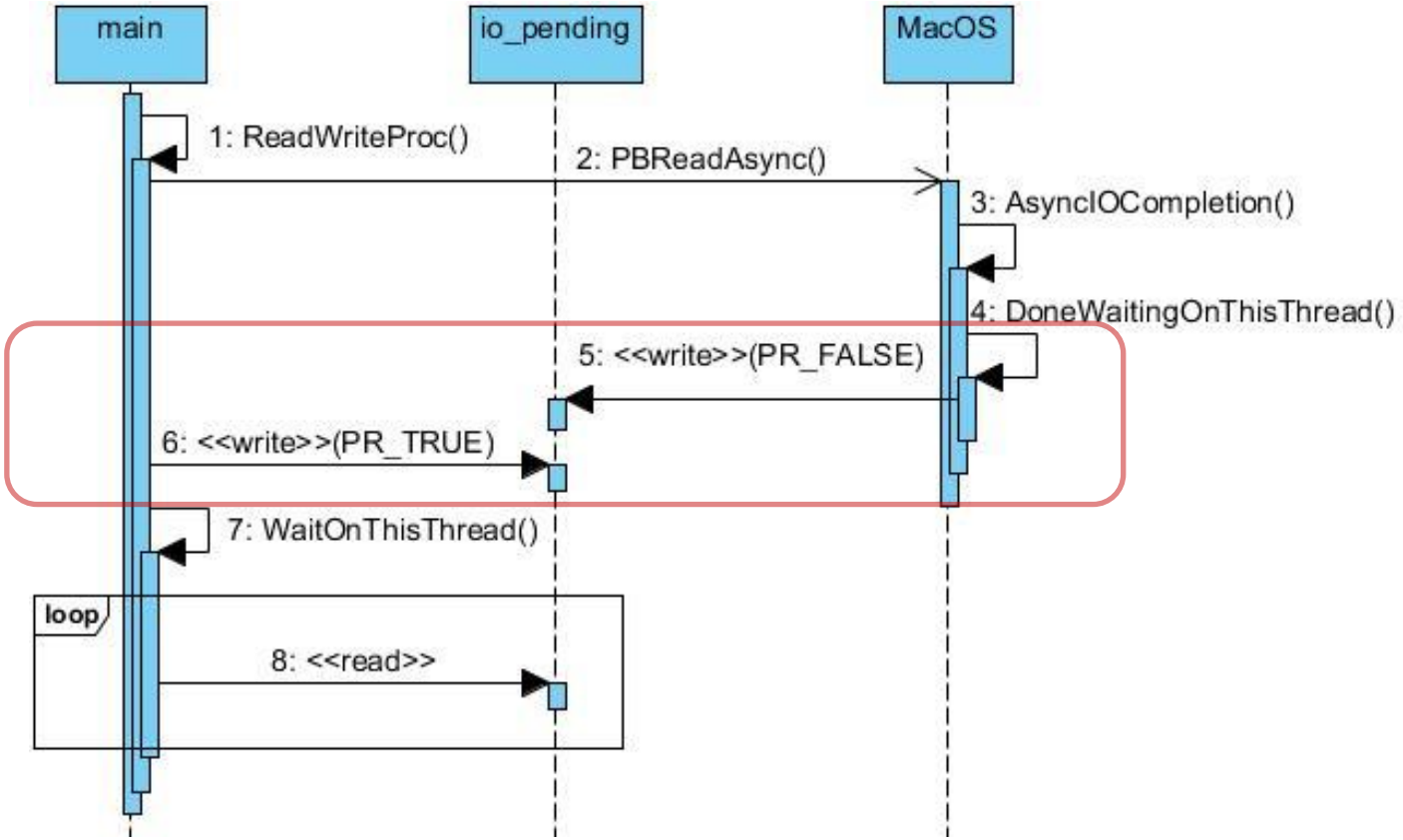
Oliver Denninger
and Martin Blersch
FZI Karlsruhe, Germany

FZI KIT Karlsruher Institut für Technologie 2013

Design Scenario: Mozilla



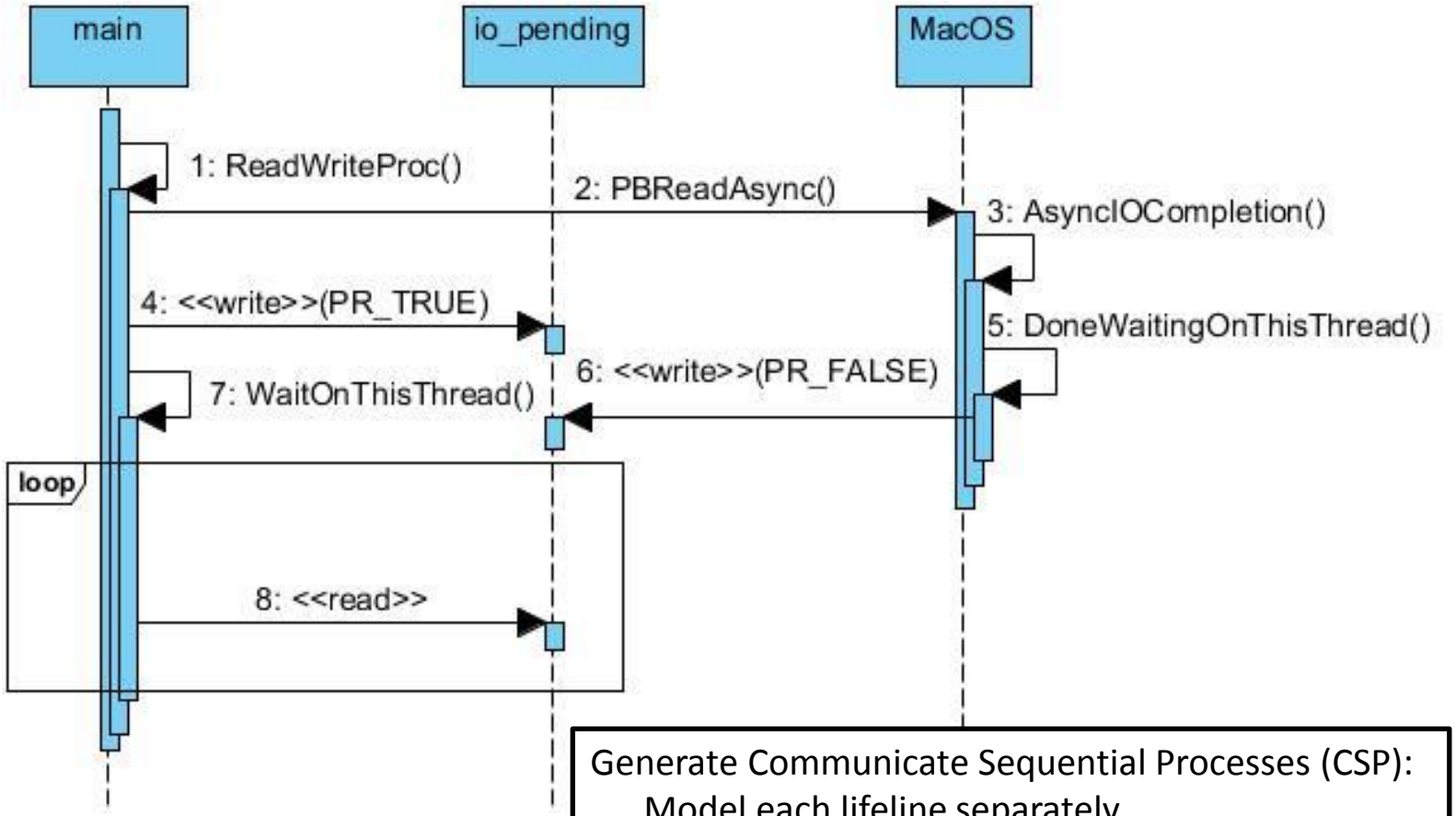
The Hidden Problem: Mozilla 97866



Defect Detection Approach

- Early detection of defects (in design)
- Formal modelling the design diagrams
- Search for concurrency defect patterns (model checking)
- Highlight suspicious parts of the diagram
- Automatic

Design Scenario: Mozilla (again)



Generate Communicate Sequential Processes (CSP):
- Model each lifeline separately
- Model method calls

CSP Model for Mozilla Example

MOZILLA97866 = (**MAIN** || **IOPENDING** || **MACOS**)

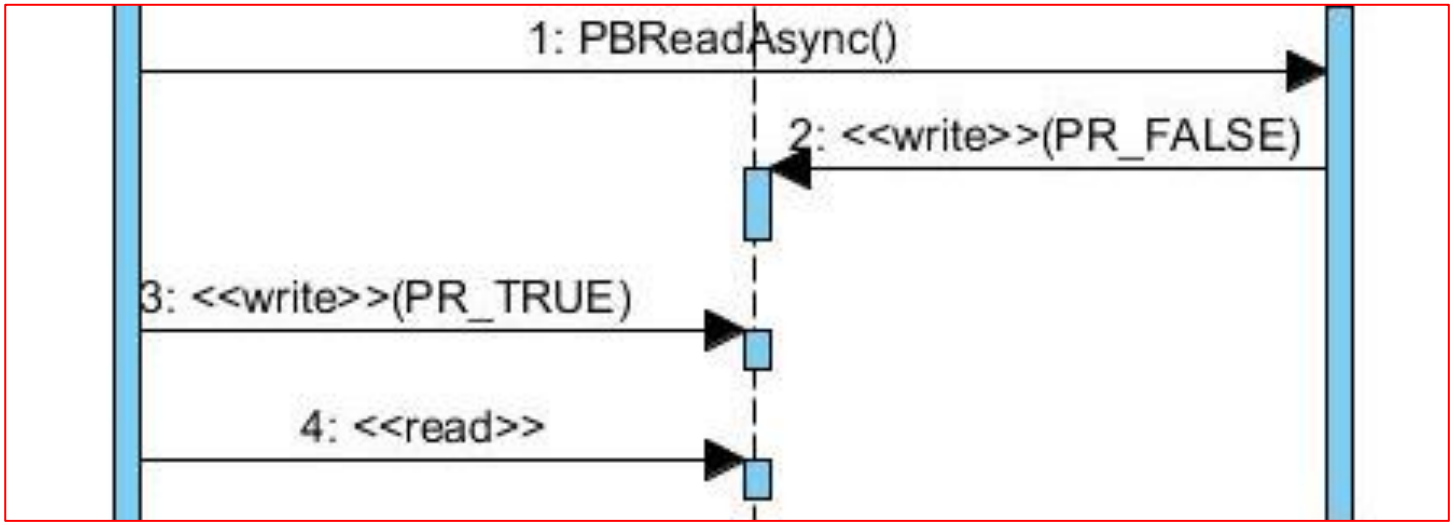
MAIN = (readwriteproc_MAIN -> *mc_MAIN_MACOS.pbreadasync* ->
wr_MAIN_IOPENDING -> waitonthisthread_MAIN -> MAIN#L)
MAIN#L = (rd_MAIN_IOPENDING -> MAIN#L
Π rd_MAIN_IOPENDING -> SKIP)

IOPENDING = (IOPENDING#S1 || IOPENDING#S2)
IOPENDING#S1 = (wr_MACOS_IOPENDING -> IOPENDING#S1)
IOPENDING#S2 = (wr_MAIN_IOPENDING -> IOPENDING#S2
| rd_MAIN_IOPENDING -> IOPENDING#S2)

MACOS = (*mc_MAIN_MACOS.pbreadasync* -> asynciocompletion_MACOS
-> donewaitingonthisthread_MACOS -> wr_MACOS_IOPENDING
-> MACOS)

Defective Trace

mc_MAIN_MACOS.pbreadasync -> wr_MACOS_IOPENDING -> wr_MAIN_IOPENDING -> rd_MAIN_IOPENDING



Patterns Catalog

Name	Patterns
Access after deletion	od_SD -> mc_P1_SD.method → STOP
	od_SD -> rd_P1_SD -> STOP
	od_SD -> wr_P1_SD -> STOP
Asynchronous wait with flag	mc_MAIN_ASYNC.method -> wr_ASYNC_FLAG -> wr_MAIN_FLAG -> rd_MAIN_FLAG -> STOP
	oa_ASYNC -> wr_ASYNC_FLAG -> wr_MAIN_FLAG -> rd_MAIN_FLAG -> STOP
Atomicity violation with one variable	rd_P1_SD -> wr_P2_SD -> wr_P1_SD -> STOP
	rd_P1_SD -> wr_P2_SD -> rd_P1_SD -> STOP
Indirect atomicity violation	mc_P1_DELEGATE.method1 -> rd_DELEGATE_SD -> mc_P2_DELEGATE.method2 -> wr_DELEGATE_SD -> mc_P1_DELEGATE.method3 -> rd_DELEGATE_SD -> STOP

Generic Pattern

Asynchronous wait pattern:

```
WOAPATTERNMETHOD = ( mc_MAIN_ASYNC.method -> wr_ASYNC_FLAG ->
                     wr_MAIN_FLAG ->  rd_MAIN_FLAG -> STOP )
```

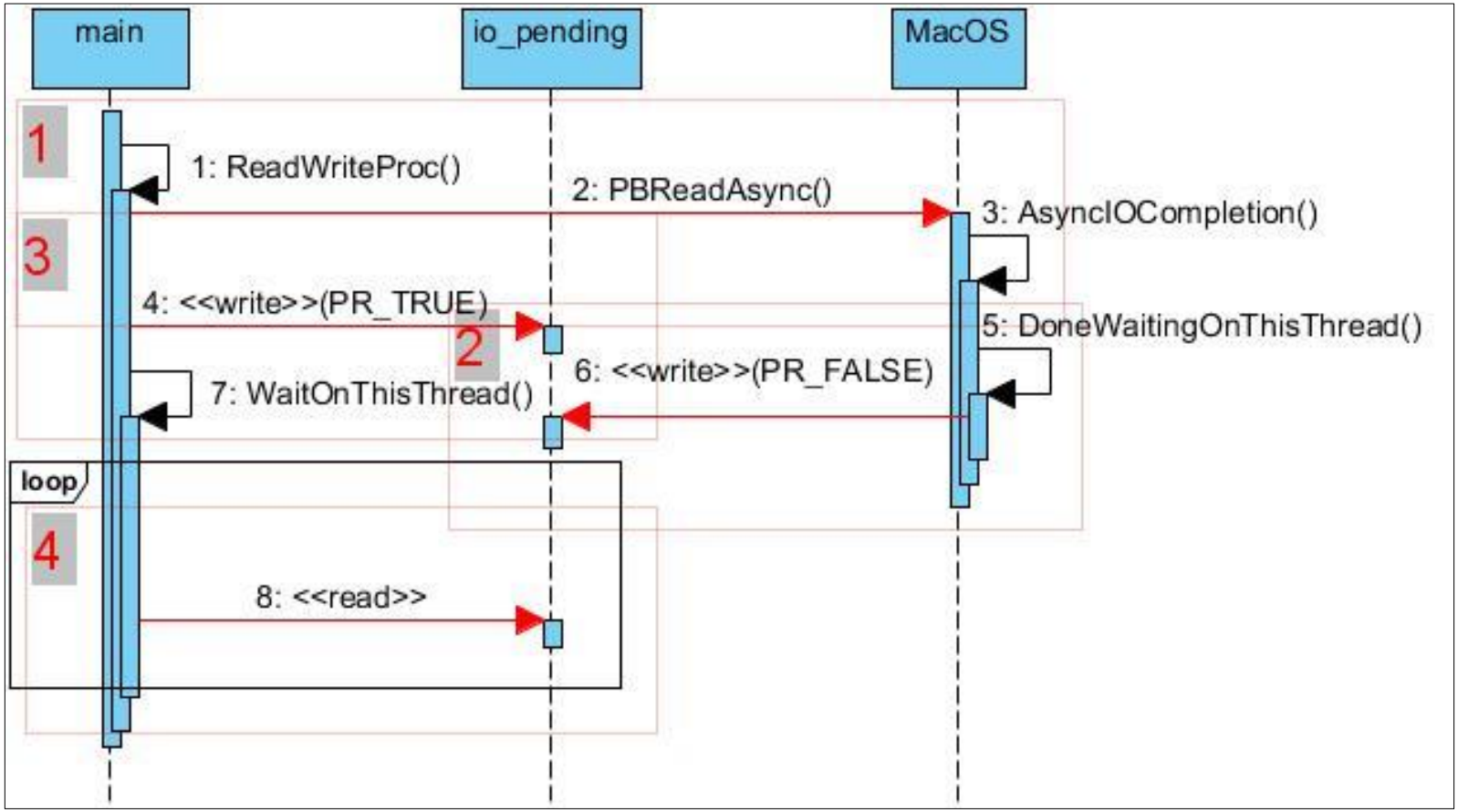
Pattern Instances

mc_MAIN_MAIN.pbreadasync -> wr_MACOS_MACOS ->
wr_MAIN_MAIN -> rd_IOPENDING_IOPENDING -> STOP

mc_MAIN_MACOS.pbreadasync -> wr_MACOS_IOPENDING ->
wr_MACOS_IOPENDING -> rd_MACOS_IOPENDING -> STOP

mc_MAIN_MACOS.pbreadasync -> wr_MACOS_IOPENDING ->
wr_MAIN_IOPENDING -> rd_MAIN_IOPENDING -> STOP

Feedback Highlighting

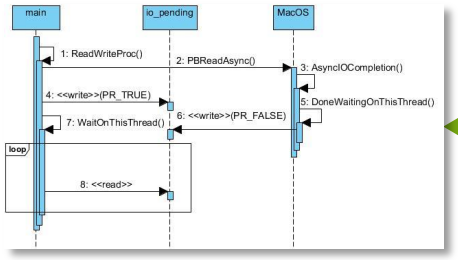


mc_MAIN_MACOS.pbreadasync -> wr_MACOS_IOPENDING -> wr_MAIN_IOPENDING -> rd_MAIN_IOPENDING

Initial Evaluation

Scenario	Bug Pattern	Detection Time
Mozilla 97866	Asynchronous wait with flag 2x Atomicity violation	0.8 seconds
PBZIP2	Access after deletion	2.1 seconds
MySQL 3596	Indirect atomicity violation	0.8 seconds
Mozilla 19421	Access after deletion	0.6 seconds
Mozilla 73291	10x atomic violation	1.8 seconds
Mozilla 73291 Synchronized	*-correct synchronisation-*	2.4 seconds

Tool Chain



mc_MAIN_MACOS.pbreadasync -> wr_MACOS_IOPENDING -> wr_MAIN_IOPENDING -> rd_MAIN_IOPENDING -> STOP

Model Generation

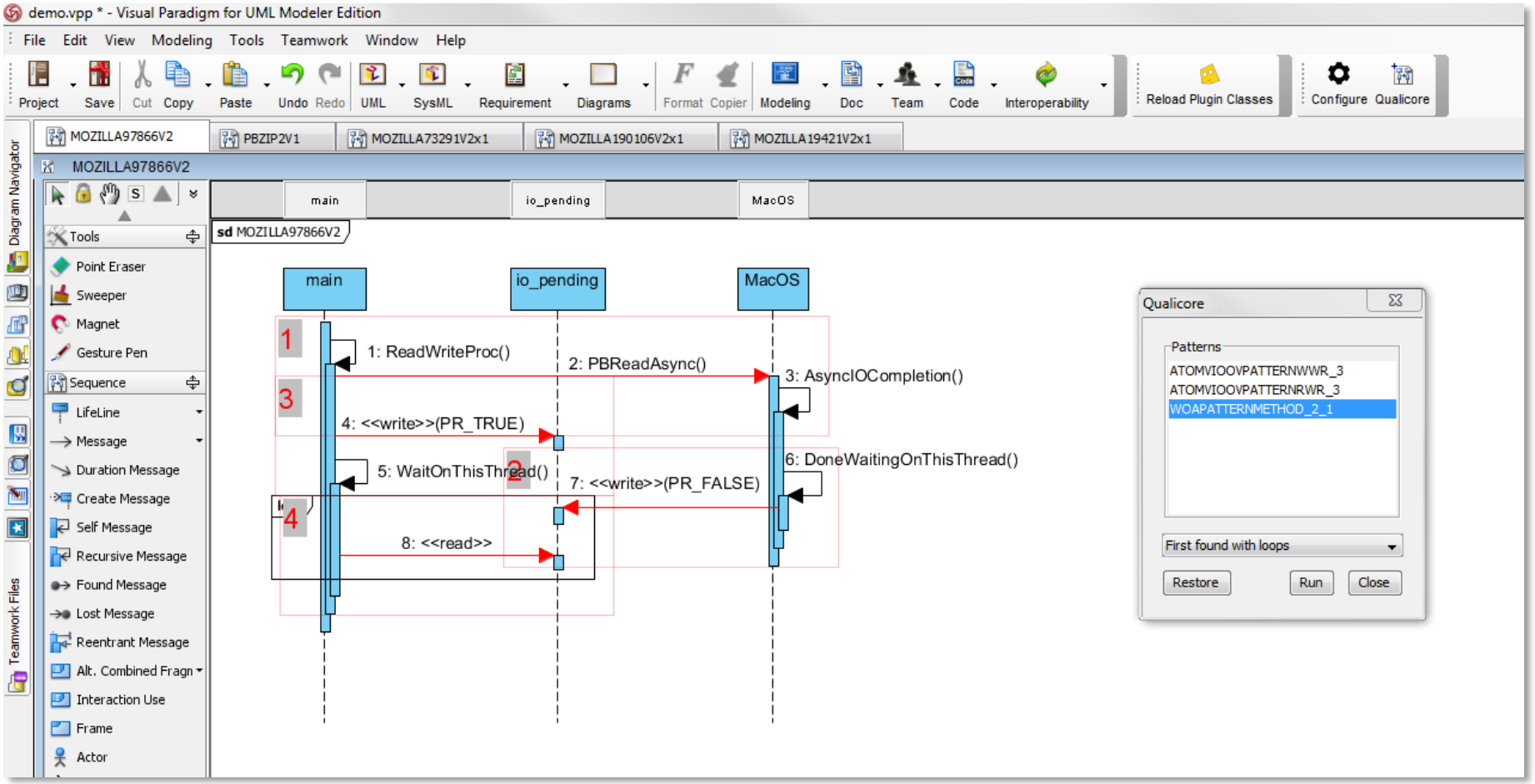
Model Checker

MOZILLA97866 = (**MAIN** || **IOPENDING** || **MACOS**)
MAIN = (readwriteproc_MAIN -> mc_MAIN_MACOS.pbreadasync -> wr_MAIN_IOPENDING -> waitonthisthread_MAIN -> MAIN#L)
MAIN#L = (rd_MAIN_IOPENDING -> MAIN#L Π rd_MAIN_IOPENDING -> SKIP)
IOPENDING = (IOPENDING#S1 || IOPENDING#S2)
IOPENDING#S1 = (wr_MACOS_IOPENDING -> IOPENDING#S1) …

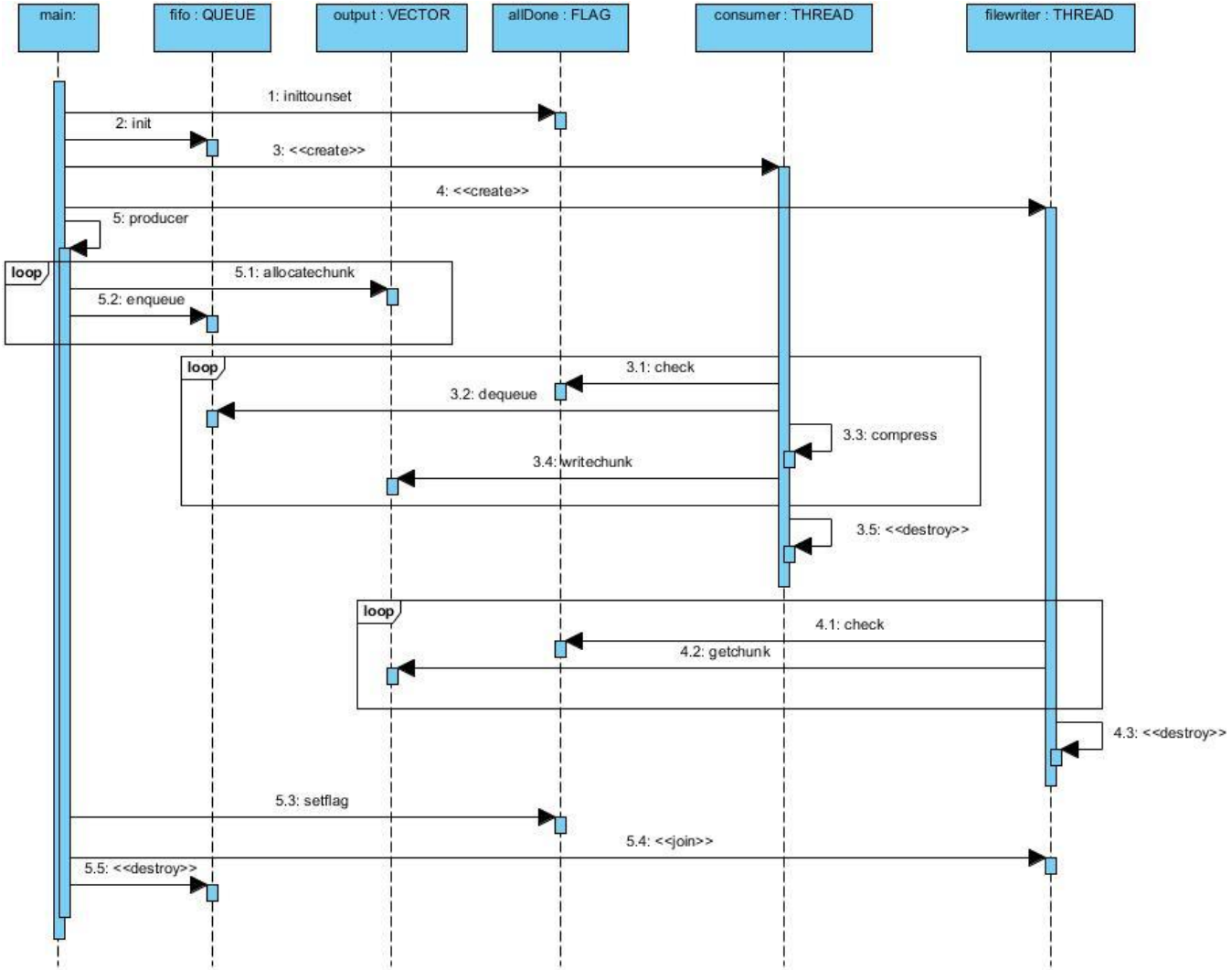
Pattern Matching

Pattern Catalog

UML Tool Plug-in



PBZIP - Diagram



PBZIP – Model

PBZIP2V1 = (**MAIN** || **CONSUMER** || **FILEWRITER** || **FIFO** || **OUTPUT** || **ALLDONE**)

MAIN = (mc_MAIN_ALLDONE.inittounset -> mc_MAIN_FIFO.init -> oc_CONSUMER -> oc_FILEWRITER -> producer_MAIN -> MAIN#L)

MAIN#L = (mc_MAIN_OUTPUT.allocatechunk -> mc_MAIN_FIFO.enqueue -> MAIN#LI Π mc_MAIN_OUTPUT.allocatechunk ->
mc_MAIN_ALLDONE.setflag -> oj_MAIN_FILEWRITER -> od_FIFO -> SKIP)

MAIN#LI = (mc_MAIN_FIFO.enqueue -> MAIN#L)

CONSUMER = (CONSUMER#S1 || CONSUMER#S2)

CONSUMER#S1 = (oc_CONSUMER -> CONSUMER#S1) CONSUMER#S2 = (mc_CONSUMER_ALLDONE.check -> mc_CONSUMER_FIFO.dequeue ->
compress_CONSUMER -> mc_CONSUMER_OUTPUT.writechunk -> CONSUMER#S2I
Π mc_CONSUMER_ALLDONE.check -> od_CONSUMER -> SKIP)

CONSUMER#S2I = (mc_CONSUMER_FIFO.dequeue -> compress_CONSUMER -> mc_CONSUMER_OUTPUT.writechunk -> CONSUMER#S2)

FILEWRITER = (FILEWRITER#S1 || FILEWRITER#S2)

FILEWRITER#S1 = (oc_FILEWRITER -> FILEWRITER#S1L)

FILEWRITER#S1L = (mc_FILEWRITER_ALLDONE.check -> mc_FILEWRITER_OUTPUT.getchunk -> FILEWRITER#S1LI
Π mc_FILEWRITER_ALLDONE.check -> od_FILEWRITER -> SKIP)

FILEWRITER#S1LI = (mc_FILEWRITER_OUTPUT.getchunk -> FILEWRITER#S1L)

FILEWRITER#S2 = (oj_MAIN_FILEWRITER -> SKIP)

FIFO = (FIFO#S1 || FIFO#S2)

FIFO#S1 = (mc_MAIN_FIFO.init -> FIFO#S1 | od_FIFO -> SKIP | mc_MAIN_FIFO.enqueue -> FIFO#S1)

FIFO#S2 = (mc_CONSUMER_FIFO.dequeue -> FIFO#S2)

OUTPUT = (OUTPUT#S1 || OUTPUT#S2 || OUTPUT#S3)

OUTPUT#S1 = (mc_MAIN_OUTPUT.allocatechunk -> OUTPUT#S1)

OUTPUT#S2 = (mc_FILEWRITER_OUTPUT.getchunk -> OUTPUT#S2)

OUTPUT#S3 = (mc_CONSUMER_OUTPUT.writechunk -> OUTPUT#S3)

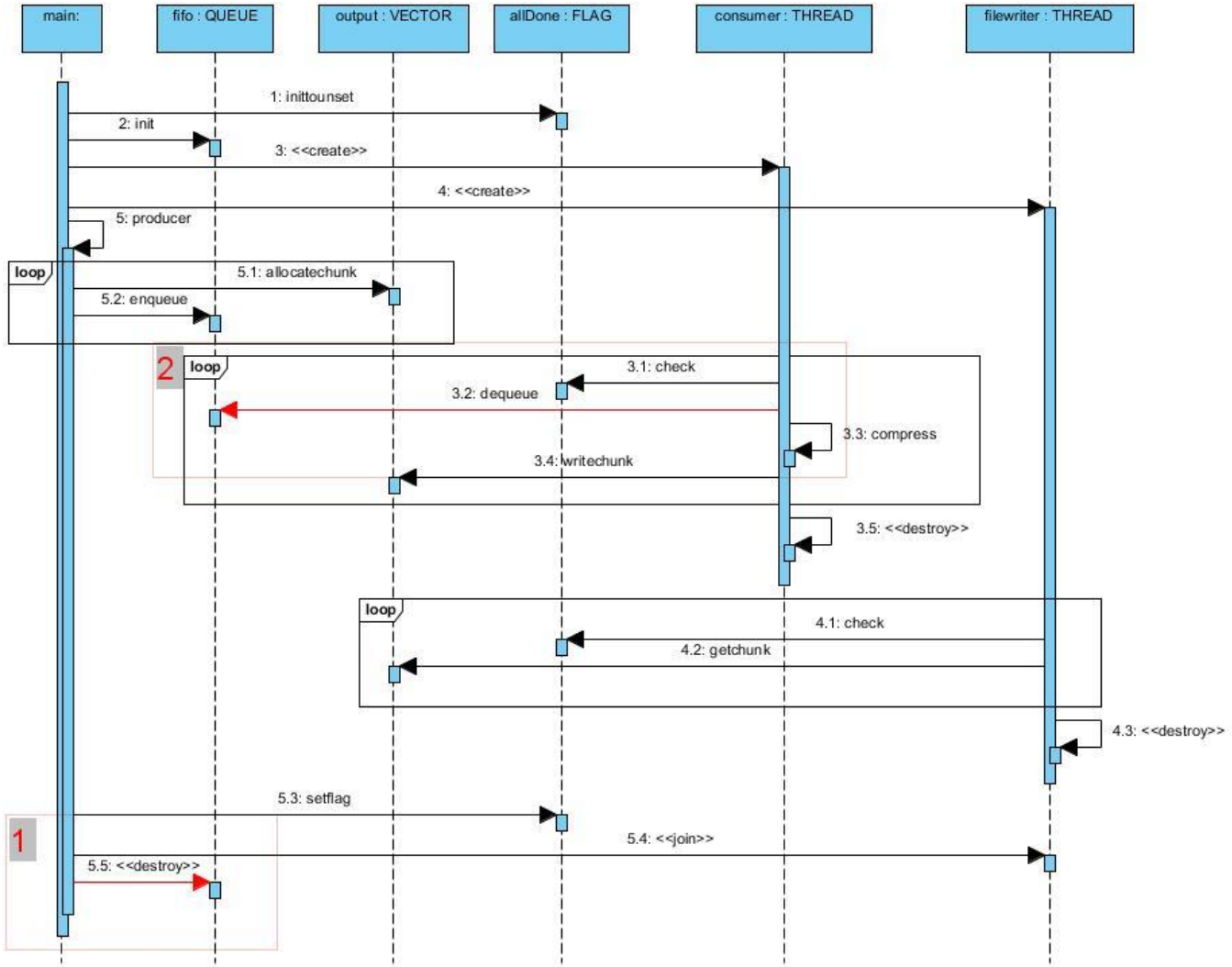
ALLDONE = (ALLDONE#S1 || ALLDONE#S2 || ALLDONE#S3)

ALLDONE#S1 = (mc_MAIN_ALLDONE.setflag -> ALLDONE#S1 | mc_MAIN_ALLDONE.inittounset -> ALLDONE#S1)

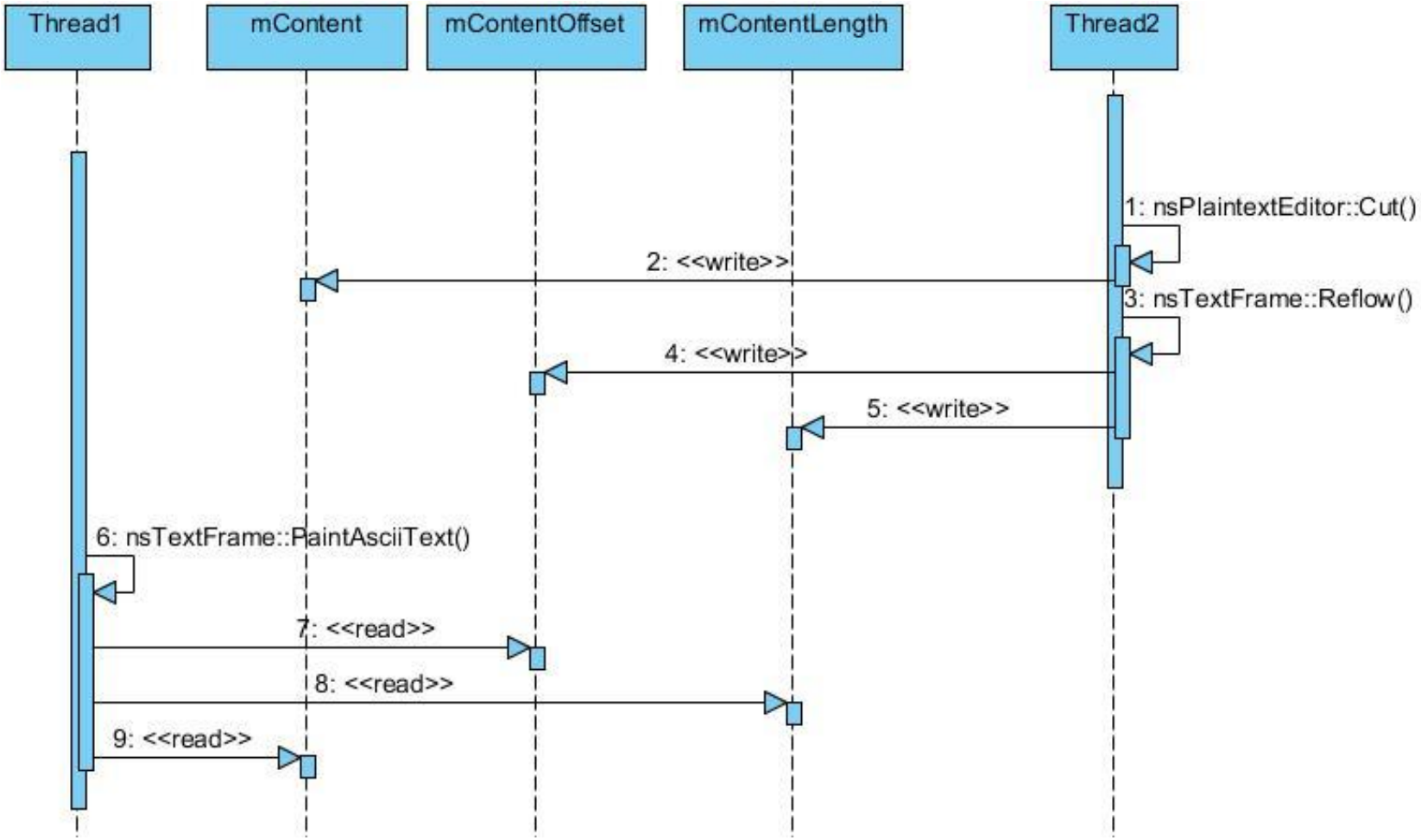
ALLDONE#S2 = (mc_FILEWRITER_ALLDONE.check -> ALLDONE#S2)

ALLDONE#S3 = (mc_CONSUMER_ALLDONE.check -> ALLDONE#S3)

PBZIP – Highlighted Diagram



Mozilla 73291 - Diagram



Mozilla 73291 – Model

MOZILLA73291V2X1 = (**THREAD1** || **THREAD2** || **MCONTENT** || **MCONTENTOFFSET** || **MCONTENTLENGTH**)

THREAD1 = (nstextframepaintasciitext_THREAD1 -> rd_THREAD1_MCONTENTOFFSET -> rd_THREAD1_MCONTENTLENGTH -> rd_THREAD1_MCONTENT -> SKIP)

THREAD2 = (nsplaintexteditorcut_THREAD2 -> wr_THREAD2_MCONTENT -> nstextframereflow_THREAD2 -> wr_THREAD2_MCONTENTOFFSET -> wr_THREAD2_MCONTENTLENGTH -> SKIP)

MCONTENT = (MCONTENT#S1 || MCONTENT#S2)

MCONTENT#S1 = (rd_THREAD1_MCONTENT -> MCONTENT#S1)

MCONTENT#S2 = (wr_THREAD2_MCONTENT -> MCONTENT#S2)

MCONTENTOFFSET = (MCONTENTOFFSET#S1 || MCONTENTOFFSET#S2)

MCONTENTOFFSET#S1 = (rd_THREAD1_MCONTENTOFFSET -> MCONTENTOFFSET#S1)

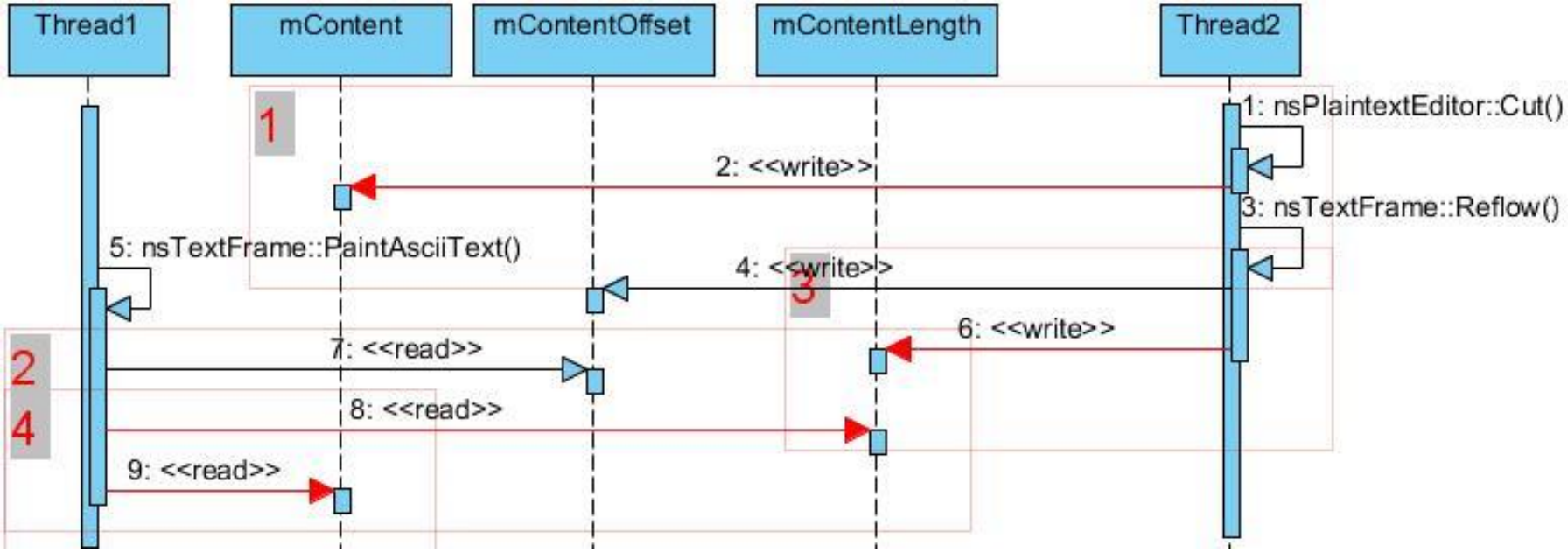
MCONTENTOFFSET#S2 = (wr_THREAD2_MCONTENTOFFSET -> MCONTENTOFFSET#S2)

MCONTENTLENGTH = (MCONTENTLENGTH#S1 || MCONTENTLENGTH#S2)

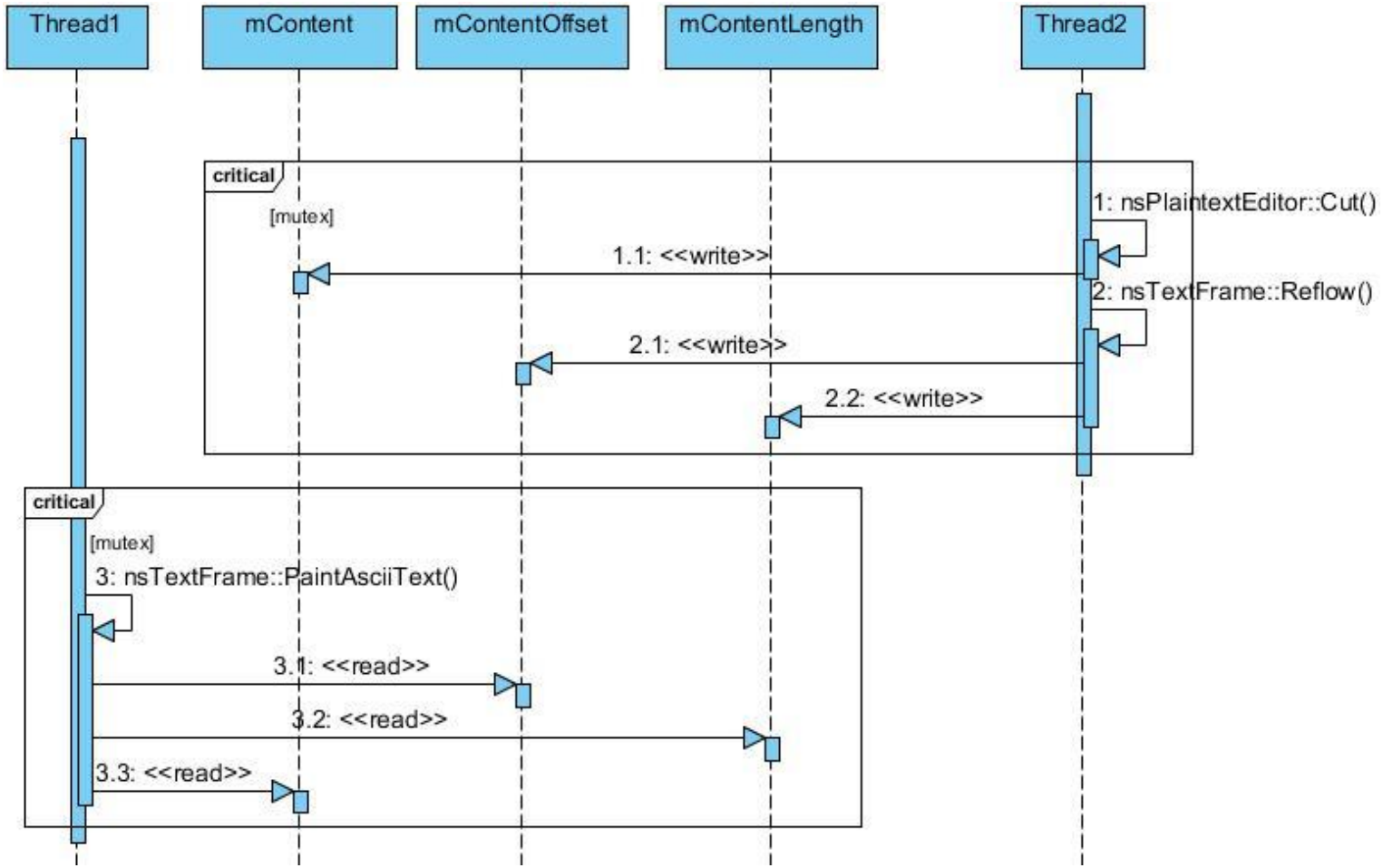
MCONTENTLENGTH#S1 = (rd_THREAD1_MCONTENTLENGTH -> MCONTENTLENGTH#S1)

MCONTENTLENGTH#S2 = (wr_THREAD2_MCONTENTLENGTH -> MCONTENTLENGTH#S2)

Mozilla 73291 – Highlighted Diagram



Mozilla 73291 Synch – Diagram



Mozilla 73291 Synch – Model

MOZILLA73291V2X1 = (**THREAD1** || **THREAD2** || **MCONTENT** || **MCONTENTOFFSET** ||
MCONTENTLENGTH || **MUTEX**)

THREAD1 = (lck_THREAD1_MUTEX -> nstextframepaintasciitext_THREAD1 ->
rd_THREAD1_MCONTENTOFFSET -> rd_THREAD1_MCONTENTLENGTH ->
rd_THREAD1_MCONTENT -> unl_THREAD1_MUTEX -> SKIP)

THREAD2 = (lck_THREAD2_MUTEX -> nsplaintexteditorcut_THREAD2 ->
wr_THREAD2_MCONTENT -> nstextframereflow_THREAD2 ->
wr_THREAD2_MCONTENTOFFSET -> wr_THREAD2_MCONTENTLENGTH ->
unl_THREAD2_MUTEX -> SKIP)

MCONTENT = (…)
MCONTENTOFFSET = (…)
MCONTENTLENGTH = (…)

MUTEX = (lck_THREAD1_MUTEX -> unl_THREAD1_MUTEX -> MUTEX
| lck_THREAD2_MUTEX -> unl_THREAD2_MUTEX -> MUTEX)