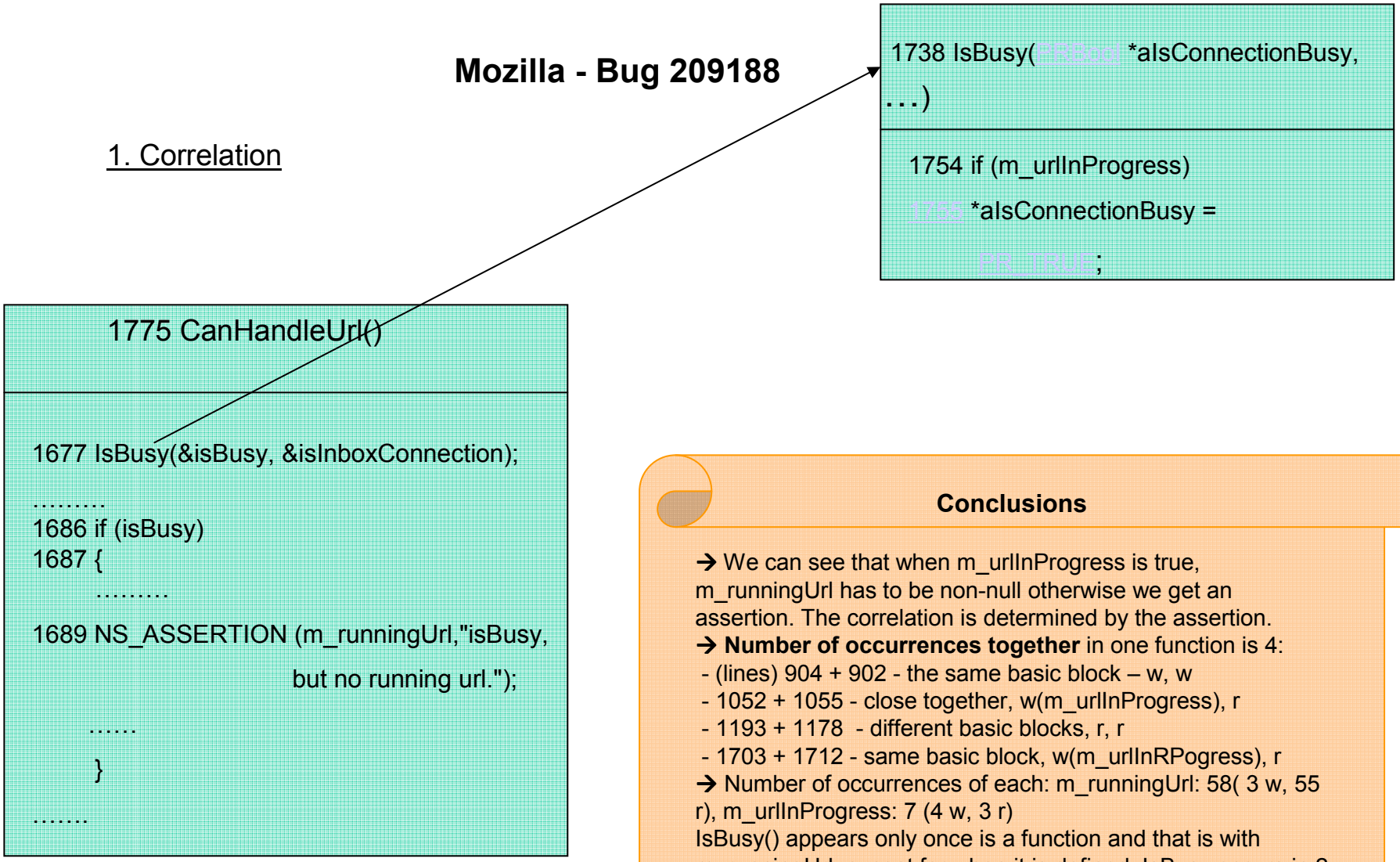
Mozilla - Bug 209188

1. Correlation



Conclusions

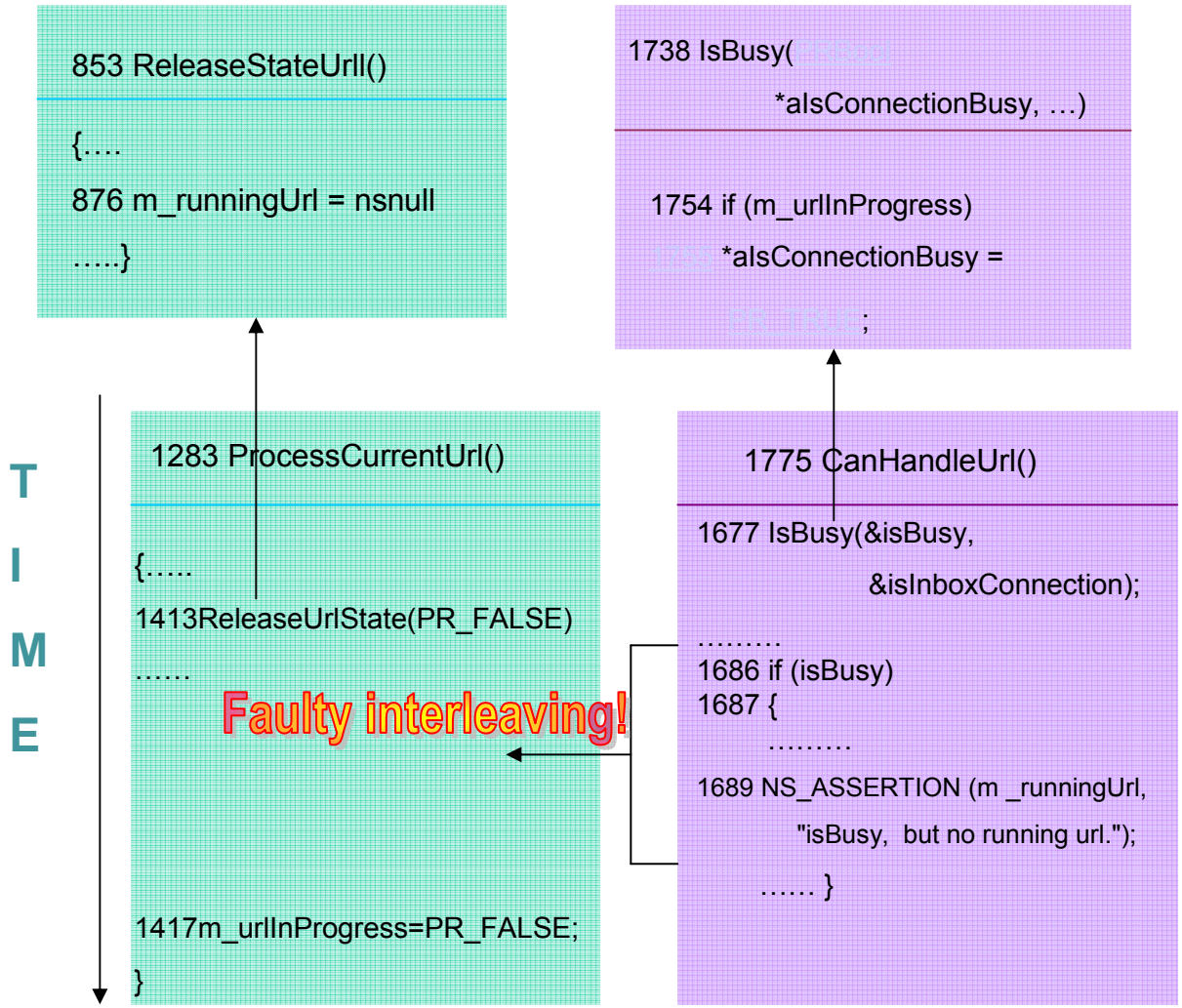
→ We can see that when m_urlInProgress is true, m_runningUrl has to be non-null otherwise we get an assertion. The correlation is determined by the assertion.
→ **Number of occurrences together** in one function is 4:
- (lines) 904 + 902 - the same basic block – w, w
- 1052 + 1055 - close together, w(m_urlInProgress), r
- 1193 + 1178 - different basic blocks, r, r
- 1703 + 1712 - same basic block, w(m_urlInRPogress), r

→ Number of occurrences of each: m_runningUrl: 58(3 w, 55 r), m_urlInProgress: 7 (4 w, 3 r)
IsBusy() appears only once is a function and that is with m_runningUrl, except for when it is defined. IsBusy occurs in 2 other functions, but it is different.
→ Also, in this example our algorithm should be able to find trans-function correlations, inline functions in others at least once.

All functions described here are in http://lxr.mozilla.org/mozilla/source/mailnews/imap/src/nsImapProtocol.cpp

Bug 209188 - continuation

2. The bug



What happens

m_runningUrl is set to null and after some time m_UrlInProgress is set to false which maintains the correct correlation. However, in the time interval between the two assignments, the correlation is not maintained and the assertion fires.

Conclusions

Here we also have to be able to see across functions to find the faulty interleaving. In this function one level of inlining is enough.

The patch makes sure that the two assignments take place atomically.

.

All functions described here are in http://lxr.mozilla.org/mozilla/source/mailnews/imap/src/nsImapProtocol.cpp