Commit Protocols

Two-Phase (2PC) and Three-Phase (3PC) Commit Protocols

Yağızcan Pançak
yagizcan.pancak@metu.edu.tr

Wireless Systems, Networks and Cybersecurity Laboratory
Department of Computer Engineering
Middle East Technical University
Ankara Turkey

March 28, 2024

Outline of the Presentation

Agenda

Distributed Commit Problem

The distributed commit problem refers to ensuring that all nodes in a distributed system either commit or abort a transaction together, preventing inconsistencies in data across the system.

One-Phase Commit Protocol

Distributed commit is often established by a coordinator. In a simple scheme, this coordinator tells all other processes that are also involved, called participants, whether to (locally) perform the operation in question. This scheme is referred to as a **one-phase commit protocol**. The drawback is that if one of the participants cannot actually perform the operation, there is no way to tell the coordinator.

Agenda

Two-Phase Commit Protocol (2PC)

The Two-Phase Commit Protocol (2PC) ensures atomicity in distributed transactions efficiently.

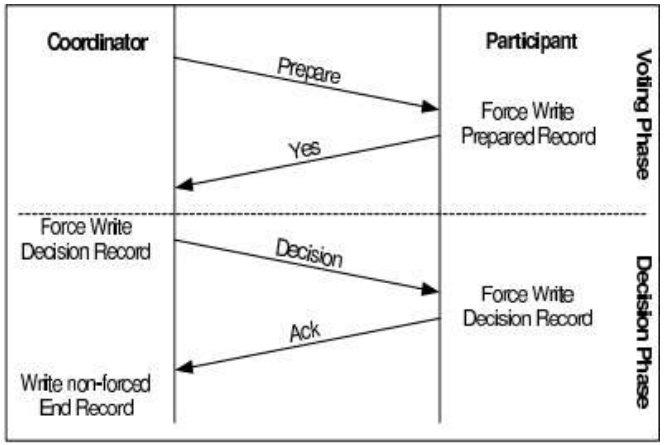

Figure: Two-Phase Commit Protocol (2PC)

- Offers a systematic approach for achieving atomicity.
- Coordinates between a coordinator and participants.
- Coordinator sends ”prepare” message, participants respond.
- Coordinator makes decision, sends ”commit” or ”abort”.
- Guarantees atomicity, minimizes inconsistencies.

Agenda

Three-Phase Commit Protocol (3PC)

The Three-Phase Commit Protocol (3PC) enhances distributed transaction atomicity with an additional phase for improved fault tolerance.

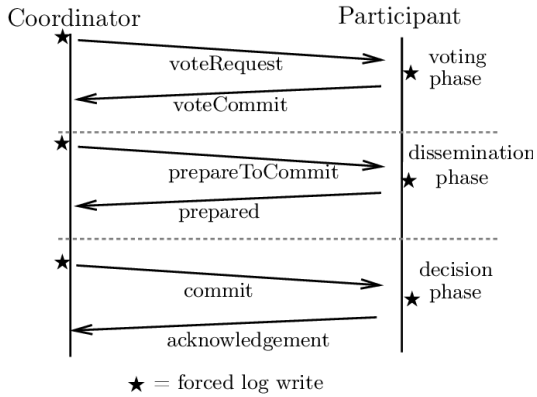

Figure: Three-Phase Commit Protocol (3PC)

- Introduces an additional phase for enhanced fault tolerance.
- Facilitates progressive commit decisions to mitigate risks.
- Coordinator solicits participant readiness to commit.
- Confirms commit readiness before final decision.
- Coordinator makes a definitive commit or abort decision.

Agenda

Motivation/Importance

Navigating Consensus: The Distributed Commit Challenge

The distributed commit problem is pivotal in ensuring agreement among system nodes on transaction outcomes in distributed systems. It poses philosophical questions about autonomy versus consensus and is addressed by models like the commit protocols, despite limitations. Its applications are widespread, but its complexity stems from reconciling consensus in fault-prone distributed environments, emphasizing the need for nuanced solutions.

Agenda

Model, Definitions

Formal definition of Commit Protocol

Coordinator: Initiates and coordinates the distributed transaction by communicating with participating nodes to ensure consistency.
Nodes: Participate in the distributed transaction by responding to the coordinator's messages, executing transactional operations, and maintaining transactional integrity.
Prepare Phase: The coordinator sends a "prepare" message to all participants, asking if they are ready to commit the transaction. Each participant replies with either a "yes" or "no" message.
Can Commit Phase: The coordinator asks participants if they can commit the transaction without actually committing it. Participants reply with "yes" or "no".
Commit Phase: If all participants reply "yes" during the "can commit" phase, the coordinator proceeds with the commit phase; otherwise, it aborts the transaction.

Background

- The Two-Phase Commit protocol ensures atomicity in distributed systems by employing two phases: prepare and commit, as introduced by Gray 1978 [1].
- The Three-Phase Commit protocol enhances the Two-Phase Commit by introducing a pre-commit phase, reducing the likelihood of blocking in case of coordinator failure, as proposed by Skeen 1981[2]

Agenda

Main Point 1: A Figure

Abstract the Major Results

Describe the key results of the paper. You may present the statements of the major theorems, but not their proofs. You will probably have to get a little technical here, but do so gradually and carefully.

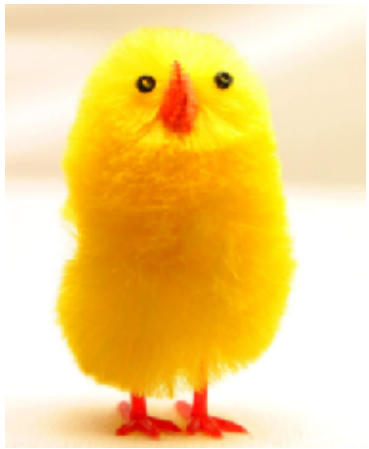

Figure: Awesome Image

An Example Distributed Algorithm

Blind Flooding

Blind flooding on an undirected graph is presented in Algorithm 1.

Implements: BlindFlooding **Instance**: cf
Uses: LinkLayerBroadcast **Instance**: lbc
Events: Init, MessageFromTop, MessageFromBottom
Needs:

OnInit: () **do**

OnMessageFromBottom: (*m*) **do**
1: **Trigger** lbc.Broadcast (*m*)

OnMessageFromTop: (*m*) **do**
2: **Trigger** lbc.Broadcast (*m*)

Algorithm 1: BlindFlooding algorithm

Main Point 2

Explain the Significance of the Results

Pause, and explain the relationships between the formal theorems that you have just presented and the informal description that you gave in the Introduction. Make it clear to the audience that the results do live up to the advance publicity. If the statements of the theorems are very technical then this may take some time. It is time well-spent.

Main Point 3

Sketch a Proof of the Crucial Results

The emphasis is on the word “sketch”. Give a very high-level description of the proofs, emphasizing the proof structure and the proof techniques used. If the proofs have no structure (in which case it may be assumed that you are not the author of the paper), then you must impose one on them. Gloss over the technical details. It is a good idea to point them out but not to explore them.

Agenda

Main Result 1

Choose **just the key results**. They should be important, non-trivial, should give the flavour of the rest of the technical details and should be presentable in a relatively short period of time. Use figures instead of tables instead of text.
Better to present 10% the entire audience gets than 90% nobody gets

Main Result 2

Try a subtitle

- Make sure your notation is clear and consistent throughout the talk. Prepare a slide that explains the notation in detail, in case that is needed or if somebody asks.
- Always label all of your axes on graphs; use short but helpful captions on figures and tables. It is also very useful to have an arrow on the side which clearly shows which direction is considered better (e.g., ”up is better”).
- If you have experimental results, make sure you clearly present the experimental paradigm you used, and the details of your methods, including the number of trials, the specific analysis tools you applied, significance testing, etc.
- The talk should contain at least a brief discussion of the limitations and weaknesses of the presented approach or results, in addition to their strengths. This, however, should be done in an objective manner – don’t enthusiastically put down your own work.

Main Result 3

- If time allows, the results should be compared to the most related work in the field. You should at least prepare one slide with a summary of the related work, even if you do not get a chance to discuss it. This will be helpful if someone asks about it, and will demonstrate your mastery of the material.
- Spell check again.
- Give for each of the x-axis, y-axis, and z-axis
- Label, unit, scale (if log scale)
- Give the legend
- Explain all symbols
- Take an example to illustrate a specific point in the figure

Agenda

Conclusions

Hindsight is Clearer than Foresight

Advices come from [**?**].

- You can now make observations that would have been confusing if they were introduced earlier. Use this opportunity to refer to statements that you have made in the previous three sections and weave them into a coherent synopsis. You will regain the attention of the non- experts, who probably didn't follow all of the Technicalities section. Leave them feeling that they have learned something nonetheless.
- Give Open Problems It is traditional to end with a list of open problems that arise from your paper. Mention weaknesses of your paper, possible generalizations, and indications of whether they will be fruitful or not. This way you may defuse antagonistic questions during question time.
- Indicate that your Talk is Over An acceptable way to do this is to say "Thank-you. Are there any questions?"[**?**]

References

[1] J. Gray, M. J. Flynn, A. K. Jones, K. Lagally, H. Opderbeck, G. J. Popek, B. Randell, J. H. Saltzer, and H. Wiehle, Eds., *Operating Systems, An Advanced Course*, ser. Lecture Notes in Computer Science. Springer, 1978, vol. 60. [Online]. Available: https://doi.org/10.1007/3-540-08755-9

[2] D. Skeen, "Nonblocking commit protocols," p. 133–142, 1981. [Online]. Available: https://doi.org/10.1145/582318.582339

How to prepare the talk?

Please read `http://larc.unt.edu/ian/pubs/speaker.pdf`

- The Introduction: Define the Problem, Motivate the Audience, Introduce Terminology, Discuss Earlier Work, Emphasize the Contributions of your Paper, Provide a Road-map.
- The Body: Abstract the Major Results, Explain the Significance of the Results, Sketch a Proof of the Crucial Results
- Technicalities: Present a Key Lemma, Present it Carefully
- The Conclusion: Hindsight is Clearer than Foresight, Give Open Problems, Indicate that your Talk is Over

Questions

THANK YOU

Commit Protocols
Two-Phase (2PC) and Three-Phase (3PC) Commit Protocols

presented by Yağızcan Pançak
yagizcan.pancak@metu.edu.tr

March 28, 2024