Progress report of a PhD Thesis – 2017-2018

PhD student: Matthieu NICOLAS

PhD year: 1^{st} year

Supervisors: Olivier PERRIN and Gérald OSTER

Laboratory: Lorraine Research Laboratory in Computer Science and its Applications (Loria)

Funding of the thesis: Doctoral contract

Subject of the PhD thesis: Efficient (re)naming in Conflict-free Replicated Data Types (CRDTs)

Short description of the subject

In order to serve an ever-growing number of users and provide an increasing volume of data, large scale systems such as data stores or collaborative editing tools have to adopt a distributed architecture. However, as stated by the CAP theorem, such systems cannot ensure both strong consistency and high availability in case of network partitions. As a result, literature and companies increasingly adopt the optimistic replication model paired with the eventual consistency model to replicate data among nodes. This consistency model allows replicas to temporarily diverge to be able to ensure high availability, even in case of network partition. Each node owns a copy of the data and can edit it, before propagating updates to others. A conflict resolution mechanism is however required to handle updates generated concurrently by different replicas.

An approach, which gains in popularity since a few years, proposes to define Conflict-free Replicated Data Types (CRDTs). These data structures behave as traditional ones, like the *Set* or *Sequence* data structures, but are designed for a distributed usage. Their specification ensures that concurrent updates are resolved deterministically, without requiring any kind of agreement, and that replicas eventually converge immediately after observing some set of updates, thus achieving *Strong Eventual Consistency*.

To achieve convergence, CRDTs proposed in the literature mostly rely on unique identifiers to reference updated elements. To generate such element identifiers, nodes often use their own identifier as well as a logical clock. Thus, regarding to how node identifiers are generated, the size of element identifiers usually increases with the number of nodes. Furthermore, element identifiers have to comply to additional constraints according to the CRDT, for example forming a dense set in case of a sequence data structure. In this case, element identifiers' size also increases according to the number of elements contained in the data structure. Therefore, the size of element identifiers is usually not bounded.

Since the size of identifiers attached to each element is not bounded, the overhead of the replicated data structure increases over time. Since nodes have to store and broadcast the identifiers, the application's performances and efficiency decrease over time. This impedes the adoption of CRDTs.

This PhD aims to address this issue. A first approach is to study identifiers proposed in the literature to list existing constraints on identifiers and their consequences on identifiers generation in order to propose more efficient specifications of identifiers. A second approach is to study this issue as a particular case of the renaming problem and to propose mechanisms to rename identifiers in order to reduce their size, still without requiring any kind of agreement between nodes.

Main results

During this first year, I focused on designing a renaming mechanism for one family of CRDTs existing in the literature : *Identifier-based Sequence CRDTs*.

Identifier-based Sequence CRDTs

Identifier-based Sequence CRDTs are data structures used to represent replicable lists. Two operations are defined to update their states : *insert* and *delete*.

The data structures attach an identifier to each inserted element. These identifiers allow us to achieve transaction-less and commutative updates by uniquely identifying each element and ordering them.

The downside of this approach is the increasing size of the identifiers. Since the identifiers are used to order the elements, they have to form a dense set so that users are always able to insert a new element between two others. However, two identifiers of the same size can be contiguous. When inserting a new element between two such identifiers, we have no other choice than to increases the size of the generated identifier to be able to generate one respecting the intended order.

Identifier-based Sequence CRDTs is a family of CRDTs particularly sensitive to the issue of ever-growing identifiers. For this reason, I focused on proposing a renaming mechanism for these data structures.

Renaming mechanism

The goal of such mechanism is to reassign shorter identifiers to each element of the data structure, without requiring any kind of agreement.

However, being a distributed system, sites can perform updates and broadcast them while another site is performing concurrently the renaming process.

Upon the reception of these concurrent updates, the site which triggered the renaming process cannot apply them to its copy. Indeed, the identifiers having changed, performing the updates would not insert the elements at the correct positions nor delete the correct elements.

To address this issue, we propose to use Operational Transformation (OT). OT proposes to transform an operation against concurrent ones to adapt its effect to the new state of the copy and preserve the user's intention.

Using this approach, we define a new operation *rename*, which reassign identifiers to each element. For simplification purposes, we make the assumption that this operation is only issuable by one node in the system, to prevent concurrent *rename* operations for now. We then propose transformation functions to adapt the *insert* and *delete* operations to the effect of a concurrent *rename* operation and conversely. Finally, we introduce an optimisation to reduce the bandwidth consumption of the *rename* operation.

This mechanism is currently being implemented in *MUTE*, the collaborative editing tool developed by the team.

Plan for next year

The plan for the next year is to 1. define a transformation function of *rename* operations against concurrent *rename* operations, to remove the previously set constraint 2. complete the implementation of the proposed renaming mechanism 3. proceed to its validation.

To validate the mechanism, the first approach is to prove that transformation functions proposed are correct. Literature in OT defines several constraints that such functions have to respect to ensure the correctness of the system. We have to determine to which of these constraints our system needs to comply with, and prove that it does.

The second approach is to run simulations to benchmark its performances. The main issue regarding this approach is the accuracy of such simulations. Indeed, the actions of the agents representing the collaborators in these simulations should be as close as possible of real world scenarios. It is then required to study the literature in the behaviour of users in collaborations to reproduce their actions faithfully.

Upon the completion of such simulations, the results will allow us to measure the impact of the mechanism on the memory usage and the performances of the application. These simulations will also be used to compare the efficiency of different strategies regarding the conditions triggering the renaming process.

Publications

[1] Matthieu Nicolas, Victorien Elvinger, Gérald Oster, Claudia-Lavinia Ignat, and François Charoy. MUTE: A Peer-to-Peer Web-based Real-time Collaborative Editor. Proceedings of 15th European Conference on Computer-Supported Cooperative Work - Panels, Posters and Demos, pages 1–4, Sheffield, United Kingdom, August 2017. EUSSET. URL https://hal.inria.fr/hal-01655438.

Project after the thesis

My current project is to pursue an academic career as a lecturer.

Scientific and professional modules validated

Scientific modules

- Réplication et cohérence des données

Professionnal modules

- Fi4 152 E Sauveteur Secouriste du Travail (SST)
- Fi4 162 C Formation à la communication orale et corporelle en milieu professionnel
- Fi4 282 Outils numériques pour la pédagogie (plateforme Arche, studio professeur)

Date and signature of the PhD student ....

Opinion of the supervisor

Name of the supervisor: ....

Opinion on this progress report: ....

Agreement for an additional year? Yes/No

Date of the defense: ....

Date and signature of the supervisor ....