

Chevrotain: A Conflict-free Replicated Data Type Key-Value Store

Alexandre Avreline
savrelin@students.cs.ubc.ca
Univeristy of British Columbia
Vancouver, British Columbia, Canada

ABSTRACT

The abstract will be here

CCS CONCEPTS

• **Networks** → **Network performance analysis**; Network reliability; • **Software and its engineering** → **Consistency**; • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics.

KEYWORDS

consistency, conflict-free replicated data types, key-value stores

1 INTRODUCTION

Testing citation [1] Testing referencing 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer metus metus, pharetra in gravida a, scelerisque vitae justo. Praesent et dui vitae nisl tincidunt varius ullamcorper facilisis eros. Fusce condimentum enim turpis, quis dapibus orci pharetra sit amet. Mauris semper mi et neque suscipit tincidunt. Sed malesuada lectus dui, nec eleifend dui hendrerit vitae. Curabitur elementum ipsum nec ex porta sodales. Cras quis eros facilisis, blandit nisi id, vehicula tellus. Sed tempor non purus et cursus. Etiam pulvinar finibus tortor, nec blandit eros dictum eu. Aenean volutpat dui sed diam aliquet, non pulvinar lectus dignissim. Praesent hendrerit nisi id lobortis suscipit. Nulla facilisi. Ut vel mauris et neque sodales gravida. Sed vel felis non ipsum ornare accumsan fermentum quis purus. Vestibulum dictum lorem vitae ligula hendrerit cursus. Sed sit amet nulla vehicula odio cursus auctor.

2 BACKGROUND

Nulla interdum imperdiet pellentesque. Vivamus consectetur neque urna, id tincidunt nibh malesuada sed. Integer sit amet neque ultricies, molestie dolor quis, imperdiet ipsum. Fusce cursus eu risus a pretium. Proin sit amet leo enim. Donec massa nulla, venenatis in porttitor at, interdum sed arcu. Curabitur aliquam orci in justo interdum posuere. Nunc ac risus sagittis est scelerisque commodo vel vel dui. Cras eros nunc, fermentum ut arcu in, molestie facilisis tortor. Sed eget ligula sed dolor maximus maximus. Fusce finibus luctus ex in iaculis. Morbi nibh magna, pretium eget eros vel, posuere varius tellus. Aliquam congue id dui viverra porttitor. Proin fermentum quam quis nunc dignissim, vel tincidunt mauris congue. Vestibulum mi felis, molestie ut ligula a, dapibus auctor ligula. Nunc hendrerit erat nec magna dapibus, vel posuere tortor vestibulum.

2.1 State-Based Convergent Replicated Data Types (CvRDT)

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

2.2 Op-Based Commutative Replicated Data Types (CmRDT)

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

3 DESIGN AND IMPLEMENTATION

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

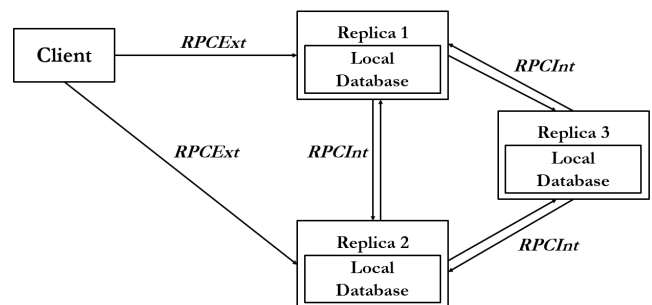


Figure 1: General System Layout

3.1 CvRDT

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet

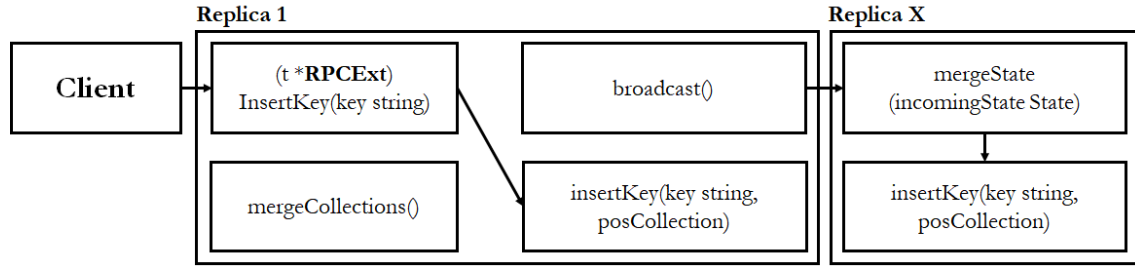


Figure 2: Typical CvRDT Workflow

Dynamic Positive Collection

"A"	{ "1", 1 }; { "2", 3 }; { "3", 6 }; { "2", 7 }
"B"	{ "4", 2 }; { "5", 4 }; { "4", 9 }

Dynamic Negative Collection

"A"	{ "2", 8 }; { "3", 10 }
"B"	{ "4", 5 };

Current Safe Tick: 8

Dynamic Positive Collection

"A"	
"B"	{ "4", 9 }

Dynamic Negative Collection

"A"	{ "3", 10 }
"B"	

Static Collection

"A"	"1", "3"
"B"	"5"

velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

7 CONCLUSION

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

REFERENCES

- [1] B. A. Davey and H. A. Priestley. 2002. *Introduction to lattices and order* (2nd ed.). Cambridge University Press, Cambridge, UK;New York, NY;.

Figure 3: Management of LWW-Sets in CvRDT

lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

5 RELATED WORK

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in velit dapibus, et volutpat nunc pellentesque. Proin id est sit amet lacus eleifend commodo. Ut eget eros molestie, luctus dolor nec, sollicitudin dolor.

6 FUTURE WORK

Nulla facilisi. Donec vitae risus et ante molestie pharetra nec vitae ex. Aliquam sit amet magna eget ante elementum rhoncus. Nam convallis cursus nisi, in auctor orci bibendum sed. Morbi risus ex, efficitur in tempus in, pretium sed ante. Proin semper neque in

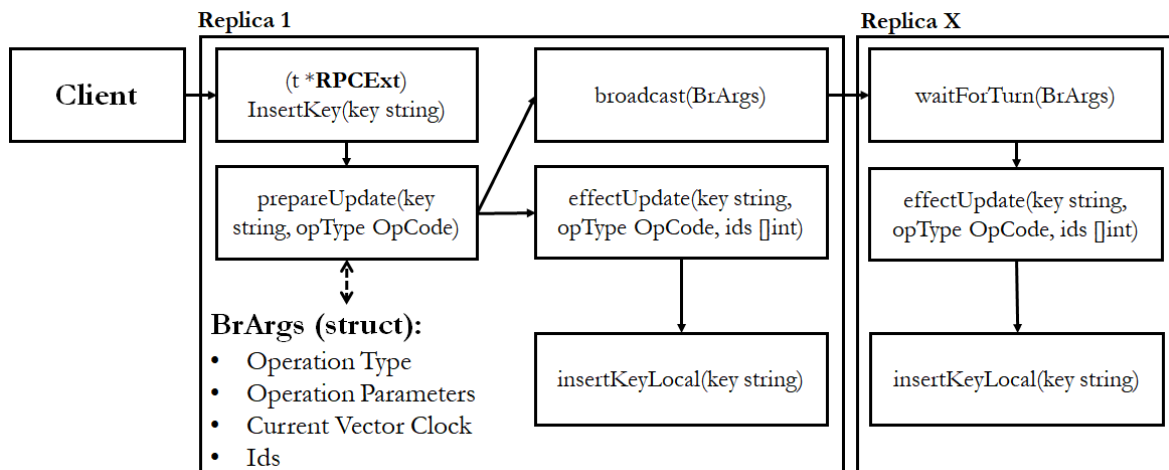


Figure 4: Typical CmRDT (Optimistic Approach) Workflow

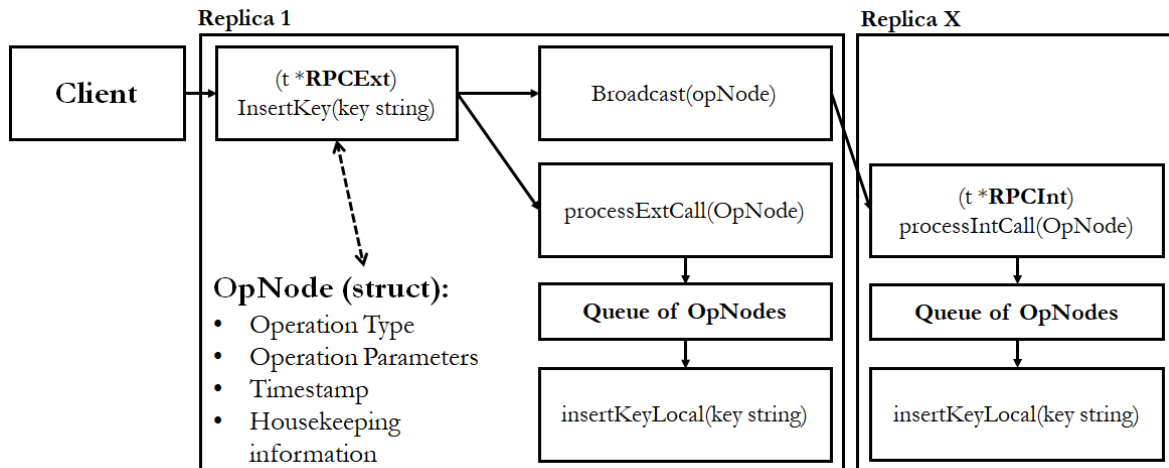


Figure 5: Typical CmRDT (Conservative Approach) Workflow



Figure 6: Queue of Operational Nodes in the CmRDT (Conservative Approach)

