RegionX Core

February 2024

Sergej Sakac
sakacszergej@gmail.com

Abstract

RegionX represents a decentralized market solution specifically designed for trading blockspace[1] within the Polkadot ecosystem. This platform provides the necessary infrastructure and implements an order book-based marketplace model, complemented by additional features and components that enable flexible and streamlined trading of blockspace. RegionX implements the components of the platform in the form of a Substrate-based parachain.

1 Introduction

Blockspace is the primary product of decentralized consensus systems, commonly known as blockchains. It encompasses the computational and storage capacity necessary to execute and finalize operations. However, not all blockchains produce the same type of blockspace. Blockspace can be differentiated based on the following three key properties:

- *Quality*, which refers to the security of the underlying consensus system. This includes how vulnerable or resistant it is to attacks.
- *Availability,* which refers to the accessibility of blockspace, encompassing factors related to supply and demand.
- *Flexibility*, which relates to the variety of operations that the underlying blockchain can support.

Polkadot stands out as a blockspace producer known for its high security and flexibility. Its consensus system is designed with validators grouped into execution cores, enabling the network to conduct multiple operations in parallel without compromising security. An analogy can be drawn between this configuration of execution cores and the architecture of a multithreaded CPU, optimizing the network's ability to process multiple tasks simultaneously.
Given that Polkadot operates on a WebAssembly (WASM) architecture, its blockspace is highly flexible, with minimal limitations on the code that can be executed by the consensus system. Polkadot can be regarded as a resilient, trustless cloud computer capable of running nearly anything that compiles to WASM.

Considering that Polkadot generates blockspace with desirable properties, efficiently using its resources is important to ensure it remains highly available. The new Agile Coretime[2] paradigm presents a novel approach to acquiring blockspace, allowing for much more granular control. RegionX is building on these advancements with the goal of making Polkadot's blockspace even more accessible.

RegionX will offer a marketplace enabling users to trade blockspace as they would any other asset. For this kind of project, it is crucial to accurately model the properties of blockspace to ensure its proper value representation in the market. This paper details the marketplace model we plan to implement with RegionX, highlighting the significance of flexible blockspace procurement and the precise reflection of blockspace value over time.

Our approach allows market accessibility through both client interfaces and automated, programmatic methods via Cross-Consensus Messaging (XCM)[3], covering all methods of blockspace procurement.
Considering these priorities, we contend that implementing this model on a parachain presents the most effective solution. This approach would grant us the required control over XCM configuration and offer broader opportunities for feature development and the exploration of further ideas.

The subsequent sections provide an in-depth explanation of Polkadot's Coretime chain, its primary market, including an analysis of its drawbacks, as well as a detailed explanation of the features and benefits of the RegionX blockspace marketplace.
It is important to note that within this document, the terms 'blockspace' and 'coretime' are used interchangeably.

2 Primary Market

The Agile Coretime RFC introduces two new allocation methods for Coretime on the Polkadot Ubiquitous Computer:

1. **Bulk Coretime**: Allocated in advance, this method involves periodic scheduling of Coretime through a specialized Coretime-chain. It is designed for users who plan their blockspace needs ahead of time.

2. **Instantaneous Coretime**: In contrast, this method allows for allocation of Coretime on the Relay-chain on an as-needed basis, right before usage and follows a block-by-block approach. It suits users requiring Coretime on short notice or for immediate tasks.

These two methods of blockspace procurement, in contrast to Polkadot's parachain slot auction model[4], improve the flexibility and efficiency of managing blockspace on the Polkadot network.
These new methods closely resemble what most cloud providers already offer. Bulk Coretime is similar to reserved instances in the cloud, while Instantaneous Coretime is comparable to spot instances.

In this model, Bulk Coretime sold on the Coretime chain is represented in the form of NFT tokens, known as 'regions'. These tokens represent ownership of blockspace for a predetermined duration, allowing their owners to assign them to particular tasks.

Every 28 days, the specialized Coretime chain launches a new sales cycle known as the, 'Bulk Sale', for these regions, thus enabling users to secure them for the upcoming time period. This periodic sales mechanism ensures that users can plan for and access the necessary blockspace in advance.
The assigned tasks will be executable on a Polkadot Core once the region becomes active in the upcoming 28-day cycle.

2.1 Properties of regions

Regions are characterized by a specific set of properties, which delineate both the duration and the quantity of blockspace to which they grant access to. Some of the most relevant properties include:

- *Begin:* Defines the start time for when a task associated with the region can begin being scheduled on a Polkadot core.
- *End:* Indicates the latest time by which a task associated with the region can be executed on a Polkadot core.
- *Core mask:* Defines the regularity at which the associated task can be scheduled on a Polkadot Core. Although it is represented as an 80-bit bitmap, it fundamentally indicates the proportion of the core allocated to the region.

These properties of regions can be modified via two distinct operations on the Coretime chain: region *partitioning* and *interlacing*.

2.2 Region partitioning

Partitioning is the operation that reduces the duration of a region, allowing regions acquired from the Coretime chain to be subdivided into segments, with a duration of less than 28 days. As a result of the partitioning process, two new regions are created, where one region starts exactly where the other one ends. The combined duration of these two regions matches the duration of the original region before it was partitioned.

2.3 Region interlacing

Interlacing is the operation that modifies the region's *Core mask*, allowing multiple tasks to simultaneously share a single Polkadot Core by taking turns. As a result of interlacing, two new regions are created, both covering the same time period but each occupying a different portion of the core they share.
Interlacing enables a single core to be shared by up to 80 different tasks.

3 Secondary Market

As described in the previous section, Coretime can be acquired either directly from the Coretime chain or from the relay chain for instances of Instantaneous Coretime. Since Instantaneous Coretime is very short-lived, it cannot be traded. However, Coretime purchased from the Coretime chain, being represented as transferable NFTs, makes the creation of a secondary marketplace both feasible and practical.

The need for a secondary marketplace arises from two main limitations associated with buying regions directly from the Coretime chain:

1. The Coretime chain exclusively offers regions representing ownership of an entire Polkadot Core for a 28-day period. This allocation is excessive for most projects, leading to inefficient usage of blockspace.

2. Regions acquired from the Coretime chain become active only at the start of the upcoming 28-day cycle. This limitation implies that any task assigned to such a region can only be scheduled on a Polkadot Core during this specified period. Consequently, projects in need of immediate or imminent blockspace access must opt for Instantaneous Coretime.

The RegionX secondary marketplace aims to address these issues by providing a platform where regions purchased from the Coretime chain can be further traded and exchanged.

Unlike the regions offered by the Coretime chain, the marketplace will feature regions with a variety of properties. This diversity is made possible because users can modify the regions before listing them for sale, utilizing the partitioning and interlacing methods available on the Coretime chain.

Additionally, in the secondary market, it will be possible not only to acquire regions for the upcoming sales period but also to purchase regions that are currently active.

This will offer teams seeking to deploy on Polkadot a significantly broader range of options for blockspace procurement.

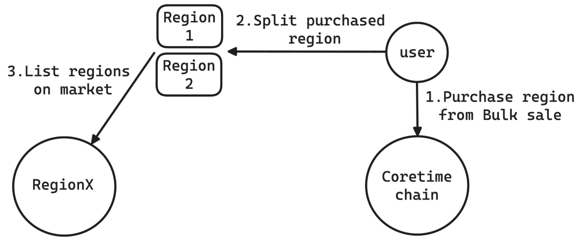

Figure 1: A possible pathway for regions entering the secondary market

Users will have the ability to purchase a region on the Coretime chain, divide it in such a way that they retain the portion sufficient for their needs, and list the remaining parts on the market.

The RegionX marketplace will be based on an order book model, enhanced by a set of features aimed at precisely reflecting the value of blockspace and facilitating highly flexible procurement.

3.1 Dynamic Pricing model

Unlike traditional NFT marketplaces, region NFTs experience a reduction in value over time due to a predetermined deadline for their use. For this reason, we are implementing a dynamic pricing model that linearly decreases the value of a region over time, culminating in a value of zero at its deadline. The region begins to depreciate in value as soon as its associated task becomes schedulable on a Polkadot core.

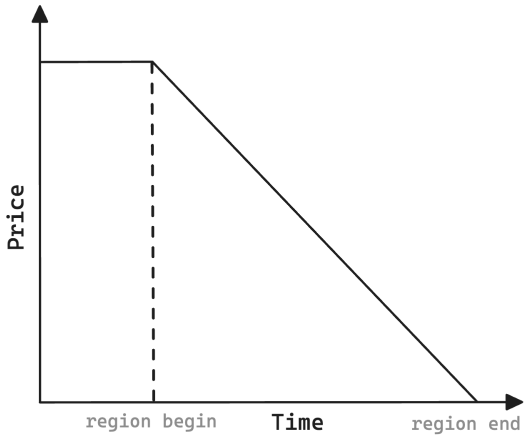

Figure 2: Price depreciation over time

Given that regions are quantized into periods called Timeslices, with one Timeslice representing a duration of 80 relay chain blocks, the dynamic pricing mechanism calculates the price of a region by multiplying its Timeslice duration by the price per Timeslice. This price per Timeslice is established by the seller when the region is listed on the market.
It is crucial to note that interlaced regions, due to their partial occupancy of a Polkadot Core, are assigned a reduced price per Timeslice. This reduction reflects their lower value, with the price per Timeslice adjusted according to the proportion of the Core the region occupies.

$$r_{price} = (r_{end} - t) * (tp * c_{occupancy})$$

Where:

- r_{end} is the timeslice at which the region concludes
- t represents the current timeslice
- tp is the cost per timeslice defined by the seller upon listing the region on the market.

- $c_{occupancy}$ represents the proportion of the Core that is occupied by the region.

4 Cross-chain

The Coretime chain, a dedicated system parachain within the Polkadot network, is responsible for the procurement and management of Coretime. It acts as the reserve chain for region NFTs. Since the Coretime chain does not support smart contract deployment, the logic for the secondary marketplace must be implemented externally.

The RegionX marketplace will be established as a dedicated parachain, incorporating the blockspace market functionality outlined in this paper in the form of substrate pallets. Additionally, it will enable other parachains within the ecosystem to access the market via XCM calls, ensuring that parachains can programmatically acquire coretime from their runtime.

Given that RegionX operates as a separate parachain from the Coretime chain, accessing the marketplace will necessitate the transfer of Coretime regions to the RegionX parachain via XCM reserve transfers.

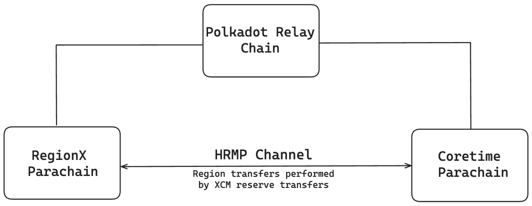

Figure 3: Cross-chain transferring regions

4.1 Access to region properties

Since the operation of the market depends on the properties of each region, it is essential not only to transfer the region NFTs but also to establish a mechanism for the RegionX chain to access and interpret the properties associated with those NFTs.

This functionality is achieved by integrating several key components. The RegionX parachain will include a 'regions' pallet, responsible for maintaining the property records of each region transferred to the parachain. Although the process for setting properties is permissionless, it requires the submission of a Merkle proof along with the data. Since being part of the Polkadot network, RegionX can access the Coretime chain's state root, which allows it to verify the provided Merkle proof.

The verification process conducted by the regions pallet:

$$Verify(\rho,\ P,\ \pi) \in \{true,\ false\}$$

Where:

- ρ is the state root of the Coretime chain.
- P represents the properties associated with a region.
- π is the merkle proof provided by the offchain worker.

However, to alleviate the responsibility from users for setting the properties of regions, an offchain worker(OCW)[5] will be introduced to the RegionX node.

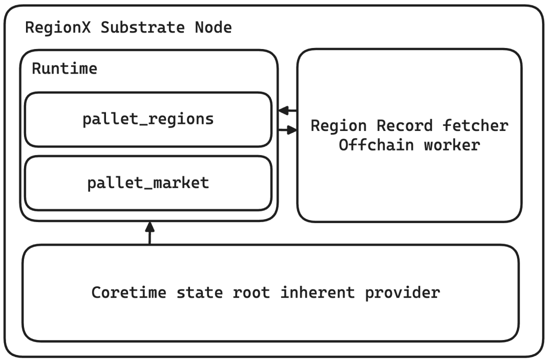
RegionX Substrate Node
Runtime
pallet_regions
pallet_market
Region Record fetcher Offchain worker
Coretime state root inherent provider

Figure 4: RegionX Substrate Node architecture

This OCW will be responsible for fetching and submitting the properties from the Coretime chain. It will be activated each time a new region is cross-chain transferred to the RegionX parachain, responsible for supplying the associated property records.

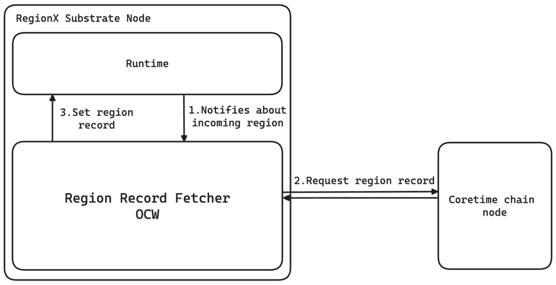

Figure 5: Fetching Region Records

Given that the properties of regions transferred to the RegionX chain cannot be altered without first transferring them to the Coretime chain, we can be certain that these properties remain unchanged while the region resides on the RegionX parachain.

While running the offchain worker does not provide additional incentives, it is important to note that collators benefit from having more regions listed on the market, as this could lead to more transaction fees being collected. Additionally, anyone can manually submit the properties of a region in case it doesn't get set by the offchain workers.

4.2 XCM interface

Although initially Coretime might be procured manually through user interfaces, in the future, much of the engagement on the marketplace will come directly from parachain runtimes within the Polkadot network.

For this reason, it is important to enable access to the market through programmatic means via XCM. The RegionX parachain will allow direct market access by exposing its marketplace pallet, ensuring it is accessible through the XCM Transact instruction. In this manner, purchasing Coretime on the secondary market involves transferring tokens to the parachain's sovereign account on RegionX, after which a purchase instruction can be submitted via XCM. Upon successfully purchasing a region in the market, the parachain will be able to transfer it over to the Coretime chain, where it can then be assigned to a specific task.

Selling Coretime involves the reverse process. First, the parachain purchases a region in the bulk sale, then it performs a cross-chain transfer to the RegionX parachain, where the region is subsequently listed on the market. Once the region is successfully sold, the parachain receives the payment from the sale into its sovereign account.

5 Conclusion

As Polkadot continues to evolve alongside its expanding development ecosystem, the utilization of its blockspace is expected to increase. Consequently, managing blockspace effectively becomes crucial to maintaining its high availability. Therefore, it is essential to establish flexible and adaptable methods for blockspace procurement.

In conclusion, the RegionX project is designed to create a blockspace marketplace within the Polkadot ecosystem, utilizing a pricing model that reflects the depreciating nature of blockspace. It will afford users the flexibility to sell surplus blockspace or purchase additional blockspace as their needs dictate.
Such a marketplace will be key for encouraging more development on Polkadot, as it will make it easier for projects to get started by reducing the barriers to entry for deployment.

References

[1] Robert Habermeier. Blockspace over Blockchains: https://www.rob.tech/polkadot-blockspace-over-blockchains/
[2] Agile Coretime: https://github.com/polkadot-fellows/RFCs/blob/main/text/0001-agile-coretime.md
[3] Polkadot Cross-Consensus Message (XCM) Format: https://github.com/paritytech/xcm-format
[4] Parachain slot auction model: https://wiki.polkadot.network/docs/learn-auction
[5] Offchain workers: https://docs.substrate.io/learn/offchain-operations/