

RYBEON

WHITE PAPER

CHAPTER 1

WHAT IS RYBEON

Before the advent of modern-day smartphones, existing phones looked perfect. Steve Jobs and his team then looked at phones from a perspective of what they should be like/from a perspective of the best way they should exit/from a perspective of perfection, and we were ushered into a new and astonishing era of smartphones. Perfection is beautiful and should be striven for.

Rybeon is a suite (of technologies) that gives software engineers the capacity to carry out perfect software engineering. Rybeon is an attempt to bring perfection to software engineering.

Composition

Rybeon is a collection of a general-purpose programming language (Rybe) and some brilliant supporting technologies like reapplicable (reusable) code infrastructure, language translation softwares, language migration technologies, etc.

Audience

Rybeon is meant for all types of entity that engineer softwares: governments, organizations, companies, startups, small businesses, groups, individuals, AIs, aliens, etc.

Usage

Rybe will see applications in system software engineering, application software engineering, cloud software engineering, game engineering, AI engineering, etc.

CHAPTER 2

THE GENESIS

The creator of Rybeon (Brian Ibrahim Qamardeen) is a Software Engineer who had used a total of 12 programming languages (C, PHP, JavaScript, *SQL*, C++, Java, Python, Ruby, Lua, Kotlin, Golang, and Rust).

Often, when he picked up a new programming language, the language taught him new programming philosophies, some of which he sometimes found phenomenal; we will call this type of thing an epiphany.

After picking up Golang and Rust and experiencing epiphanies again, he started to wonder if there were more beautiful programming philosophies.

He felt there should be more of them out there.

As a perfectionist, Brian started to wonder how he could acquire all the beautiful programming philosophies that possibly exist.

He came to the realization that, if he wants to acquire them all, he has to start approaching programming according to how it should be done, and not how existing languages encourage it to be done.

When Brian started to approach programming according to how it should be done, he started to see many ways programming can be significantly better, compared to mainstream approaches.

Brian started trying to adopt his newly discovered programming approach to his day-to-day programming tasks. Because existing languages were not designed based on his newly discovered approach, it was very difficult to adopt the approach, using any of the existing languages.

Because Brian could not adopt his approach using any of the existing languages, he decided to create a language that enables him to do so. This decision gave birth to Rybeon in October 2019.

However, The Initial Idea Has Grown

Although Brian's initial intention was just to create a programming language, he saw that it would be wise to morph the project into a bigger idea: a suite of programming technologies (a language and supporting technologies like infrastructure that facilitate code reapplication, technologies that can help interested entities convert their codebase from other languages to Rybe, etc).

CHAPTER 3

PRESENT STATE

As of August 2021, the design of Rybe has undergone 6 iterations already. The present design (the 7th iteration) is close to completion. This iteration is a release candidate.

Hopefully, before 2021 runs out, a sequel release will become available.

CHAPTER 4

WHAT RYBE LOOKS LIKE

The best way to show what Rybe looks like is probably to write a "hello world" program with it, but, because the design is yet to be completed, this is impossible.

However, I will still give a glimpse into what the language looks like, by creating an information (variable) representing some texts:

```
s1.1:G1.1
=====
eftd| crte +Information
eftd| +Information: asqn [!, [u: Hello world!]]
eftd| +Information: asqn [!, [u: This is beautiful!]]
eftd| +Information: aqsn [!, [u: This is the language of the future!]]
```

CHAPTER 5

QUESTIONS YOU MAY HAVE

1: How Similar is Rybe to Existing Languages?

Rybe is very different from any mainstream programming language that has ever existed.

2: Is Rybe an Object-oriented Language?

According to the things OOP means to most people, Rybe is not an object-oriented language. I will also like the term OOP to be distanced far away from it.

3: Will Rybeon be Enterprise-friendly?

Yes, definitely. Brian personally needs this suite to be enterprise-friendly, so he will do everything in his power to make it very enterprise-friendly.

4: What is Your Vision for Rybeon?

My vision for Rybeon is to make it a Living, Perfect, and Mature.

5: Is Rybe a Compiled Language?

No and yes.

Rybe is meant to be a language that will be executed directly by a computer: think of it as a machine language that is also easy to program with by humans.

However, at the moment, there is no computer capable of directly executing Rybe instructions; efforts will be made to make such computers come into existence. To make it possible to program existing computers in Rybe, translation softwares will be created, to translate Rybe to the languages understood by existing computers.

6: Can I Contribute to Rybeon?

Yes of course! You can! However, at the moment, monetary rewards are not offered to contributors. However, if you contribute significantly, your name will be written down as one of those who will be rewarded appropriately later in the future.

If you like the idea of Rybeon and want to contribute, shoot me an email. I will be glad to hear from you.

CHAPTER 6

CREATOR'S PROFILE

Name: Brian Ibrahim Qamardeen

Email: Qeetell@Gmail.COM

Website: Qeetell.VIP

LinkedIn: [LinkedIn.COM/in/qeetell](https://www.linkedin.com/in/qeetell)