**Submitted by:Abdul Momin.**

**1. Who and when this was invented?**

Go is a programming language developed at Google in year 2007 by Robert Griesemer, Rob Pike, and Ken Thompson. Strongly and statically typed, provides inbuilt support for garbage collection and supports concurrent programming, Announced in November 2009.

**2. Why was this created?**

Google works with a lot of data. Google had faced two big problems. One of them is compile time. Some project took hours to do a clean build. To reduce the compile time GOLANG was designed. Another problem is string processing, Google analyzes a lot of web pages, which are text files. They do a lot of string functions into GO, Garbage collecting makes strings in GO simple to think about and efficient in ways some other string libraries.

**3. Who are using this? Companies & individuals?**

Companies currently using GO throughout the world.

* MROffice - Built their “\*MROffice Dialer” telephony solution using Go.
* Clarity Services, Inc. - Using Go for event based post-processing of credit applications.
* Cablenet Communication Systems Ltd - In-house Provisioning System developed in Go.
* Feedbooks - Use Go and mgo to serve more than a million book covers a day.
* Anchor - Developed and uses the Orchestra distributed execution framework using Go.
* Carbon Games - Using Go for their backend server stuff.
* vaba Software - Rewrote their message and storage engines in Go.
* Institute for Systems Biology - Developed the Golem distributed computational analysis system.
* Second Bit - Using Go to power their 2cloud service.
* Meetinarts - Built their whole system in Go.
* Apcera
* SmugMug
* Novartis - Uses an internal application to access their inventory system of cell lines written in Go.
* Airbrake - Migrated from Ruby to Go.
* Unicorn Enterprises SA - Developers of Express Go.
* Torbit
* SoundCloud
* janrain - See this HN post.
* Toshnix Systems Limited
* swirl.us
* Replicon Inc. - Developed and uses the fast-archiver tool written in Go.
* Mob Rules Games - Developing in Go the open source engine and the game code for Haunts: The Manse Macabre.
* Metrum Technologies - Using Go to gather network status and diagnostic information of over.
* Bitly
* Cloud Flare - Built their Rail gun software in Go.
* Nokia Siemens Networks - Using Go to automate baseband and RF board hardware verification testing
* Cloud Foundry, a platform as a service
* CloudFlare, for their delta-coding proxy Railgun, their distributed DNS service, as well as tools for cryptography, logging, stream processing, and accessing SPDY sites.
* CoreOS, a Linux-based operating system that utilizes Docker containers.
* Couchbase, Query and Indexing services within the Couchbase Server
* Dropbox, migrated some of their critical components from Python to Go
* Google, for many projects, notably including download server dl.google.com
* Hyperledger Fabric, an open source, enterprise-focused distributed ledger project
* MercadoLibre, for several public APIs.
* MongoDB, tools for administering MongoDB instances
* Netflix, for two portions of their server architecture
* Novartis, for an internal inventory system
* Plug.dj, an interactive online social music streaming website.
* Replicated, Docker based PaaS for creating enterprise, installable software.
* SendGrid, a Boulder, Colorado-based transactional email delivery and management service.
* SoundCloud, for "dozens of systems"
* Splice, for the entire backend (API and parsers) of their online music collaboration platform.
* ThoughtWorks, some tools and applications around continuous delivery and instant messages (CoyIM).
* Twitch.tv, for their IRC-based chat system (migrated from Python).
* Uber, for handling high volumes of geofence-based queries.
* Zerodha, for realtime peering and streaming of market data
* Canonical - Developing backend infrastructure using Go.
* BBC World News - More information about how they are using Go in this Quota post by the Director of Technology at BBC Worldwide.
* Open Knowledge Foundation - Using Go for (meta)data catalogue aggregation and linked data.
* Tinkercad Inc. - Developing a new and faster way of creating designs for your 3D printer.
* Heroku - Developers of Doozer, an open source consistent, distributed data store implemented in Go.
* Iron.io - Implemented their IronWorker scalable task queue and other services in Go.
* Numerotron Inc. - Developed their StatHat statistics and event tracking system in Go.
* ngmoco:) - Developer of games and smartphone gaming platforms. Using Go to develop the FalcoreHTTP server and Timber, a configurable logging framework.
* Argonne National Laboratory - Developing the Shock platform for computation, storage, and distribution of scientific data.
* **Bangladeshi two companies are using this language**
* AppsCode Ltd. (BD subsidiary of AppsCode Inc.)
* Telenor Health(A health service company) built USSD server with go

**4. Pros and Cons of the language?**

Pros of Golang:

* It is fast. And not only fast in the sense that programs written in it run fast when compared to other common languages; but also fast in the sense that its compiler can compile projects in the blink of an eye. You can even edit and run Go programs directly on the Web.
* It is a garbage-collected language. This puts less pressure on the developer to do memory management, as the language itself takes care of most of the grunt work needed.
* It has built-in concurrency, which allows parallelism in an easier way than is possible in other languages. Go has the concept of go routines to start concurrent work and the concept of channels to permit both communication and synchronization.
* Go has documentation as a standard feature. That makes it easier for developers to document their code and generate human-readable data out of source code comments.
* Go has a rich standard library which covers a lot of areas. In fact, Go is probably the only language that can claim to have a fully working Web server as part of its standard library.
* Go’s built-in build system is both elegant and simple. No need to mess with build configurations or make files.
* Easy to learn because it’s based on C
* GOlang is platform independent
* Has static type checking
* Fast working

Cons of Golang:

* Go is still a very young language and has a very young ecosystem. This means there aren’t many libraries for it yet, leaving developers to write libraries themselves. There is also a shortage of books and online courses on the language.
* Go is simple to the point of being superficial. Go’s simplicity is mostly superficial, and in its effort to find simplicity, it threw away decades of valuable programming language progress.
* Although Go is a high-level language, it still has low-level features such as pointer-arithmetic which does not rule out the chance of doing systems and OS programming.
* Go’s tooling is really weird, on the surface it has some really nice tools, but a lot of them, when you start using them, quickly show their limitations.
* It is still not so easy to learn Go and it’s difficult to handle errors in it.
* Haskell are missing
* Designed to make the programmer expendable

5. **Comparison with Java, Python and C:**

Programming languages are used for controlling the behavior of a machine (often a computer). Like natural languages, programming languages conform to rules for syntax and semantics.

There are thousands of programming language sand new ones are created every year. Few languages ever become sufficiently popular that they are used by more than a few people, but professional programmers may use dozens of languages in a career.

Most programming languages are not standardized by international (or national) standard, even widely used ones, such as Perl or Standard ML (despite the name). Notable standardized programming languages include ALGOL, C, C++, JavaScript (under the name ECMAScript), Smalltalk, Prolog, Common Lisp, Scheme (IEEE standard), Ada, Fortran and COBOL (SQL, HTML, XQuery and XML are also standardized).

**6. Write a summary of your opinion. Pointing why we SHOULD use this language for our project. One paragraph.**

GO is a powerful programming language created by Google 2009. Golang is a general purpose programming language which can compete with C and C++ languages, Go has a simple syntax with a minimum set of standard key words, the interesting thing with Go is, one can build complex applications, with simple programming model. We can put our effort in higher level abstract things. When we look the syntax of Golang, it eliminates lot of standard which are usual in C#, Java or C++. Go inherits many concepts from C language. To give few example, like how simple the language is, we don’t need to specify a semicolon (;) to terminate the line, no access modifiers in Golang. We can do it by declaring the function name starting with upper case or lower case. Most important things are, its compilation time is much faster than other languages, and even compiling very large projects is done in few minutes.

We should use GOlang for our project because it’s easy to learn and based on C, Most of us in our team have idea about C language so I think it’s a proper choice. Most of the syntax are easy to read and write in GOlang. And another plus point is the GO compiler compiles binaries instantly as fast a scripting language interpreter and it’s a great language for building networking services which are helpful for our project.

Research: Google, Wikipedia, GO language official website, stack overflow etc.