State of the Go SDK 2022

Robert Burke
@lostluck

BEAM SUMMIT Austin, 2022

Previously, in Beam Go

BEAM SUMMIT Austin, 2022

Previously, in the Go SDK

- Handles Batch Basics
  - Global Windowing
  - ParDos
  - Iterable Side Inputs
  - Flatten
  - CoGBKs
  - CombineFns w/Lifting
  - Partition
  - User Metrics
  - Coders
    - Standard
    - Custom Go

- Announced in 2020
  - Cross Language Transforms
  - Bounded Splittable DoFns
  - Loopback mode
  - Katas
  - State Backed Iterables
  - Reshuffle

Summer 2020

What's missing?

TODO(you?)

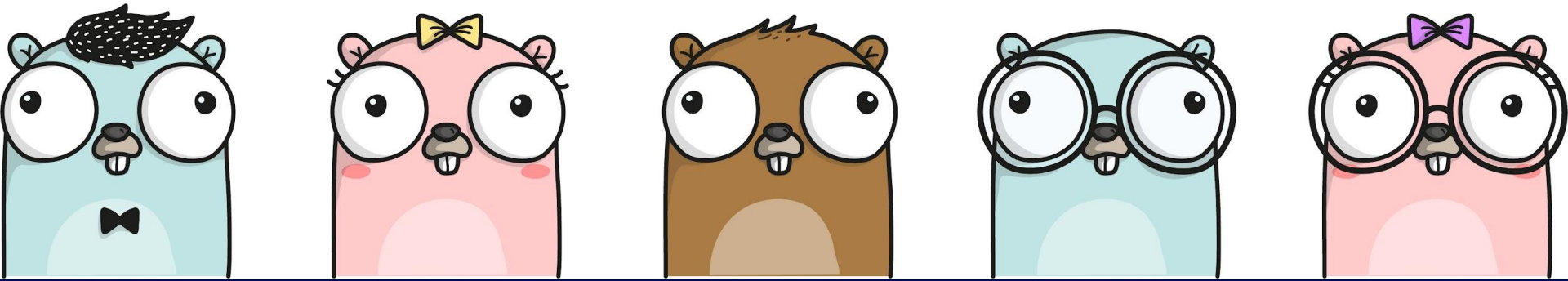
BEAM SUMMIT

- Map Side Inputs
- Advanced Windowing
- State and Timers
- DoFn Checkpointing
- Native SDF IOs
- Cross Language Wrappers
- Go SDK Expansion Service
- Testing Audit
- Go Generics
- ….?

Summer 2022

What's new?

- Exited Experimental
  - Go Module Support
  - Default Beam Schema Coding
- Pipeline Results
- Map Side Inputs
- Advanced Windowing & Triggers
- Native Streaming Support
- New Cross Language Wrappers
- Testing Audit
- Worker Status
- Go Generics for Performance
- Load Tests
- Dataflow Support



Go SDK Exited Experimental

- November 2021
- https://beam.apache.org/blog/go-sdk-release/
- Bounded Splittable DoFns
- Beam Programming Guide
- Default Schema Coding
- Go Module Support

Beam Programming Guide

https://beam.apache.org/documentation/programming-guide/

```
func init() {
  // 2 inputs and 1 output => DoFn2x1
  // Input/output types are included in order in the br
        register.DoFn2x1[context.Context, string, int](
}
```

4.2.1.1. Applying ParDo

`beam.ParDo` applies the passed in `DoFn` argument to the input `PCollection`, as

Java | Python | **Go** | Typescript

```
// ComputeWordLengthFn is the DoFn to perform on each e
type ComputeWordLengthFn struct{}

// ProcessElement is the method to execute for each ele
func (fn *ComputeWordLengthFn) ProcessElement(word stri
```

7. Data encoding and type safety

Adapt for: Java SDK | Python SDK | **Go SDK**

When Beam runners execute your pipeline, they often need to materialize the int
elements to and from byte strings. The Beam SDKs use objects called `Coder` s t
encoded and decoded.

> Note that coders are unrelated to parsing or formatting data when interacting
> formatting should typically be done explicitly, using transforms such as `ParD`

Standard Go types like `int`, `int64` `float64`, `[]byte`, and `string` and more ar
default using Beam Schema Row encoding. However, users can build and regist
available Coder functions in the coder package.

> Note that coders do not necessarily have a 1:1 relationship with types. For ex
> input and output data can use different Integer coders. A transform might ha
> and Integer-typed output data that uses VarIntCoder.

Default Schema Coding

- Schemas are automatically inferred for user defined structs.
- Uses Beam Schema Row Encoding by default
- Compact Binary format is significantly more efficient than using JSON, the previous by default.
- Similar restrictions: fields must be Exported to be encoded
- No dynamic Row type at this time.

Go Module Support

import “github.com/apache/beam/sdks/v2/go/pkg/beam”

- go.mod rooted in *sdks/* folder.
- v2.40.0 has a minimum language version of go 1.18
  - Doesn’t prevent users from adopting newer Go versions themselves
- Ensures SDK developers, SDK Users, and Repo Test automation all use the same dependency versions

Do More

- Triggers
- Cross Language Transforms
- Native Streaming
- Map Side Inputs
- Pipeline Results

Triggers

- Configure Aggregation Behavior
- Combines with Fixed, Sliding, and Session interval windowing strategies
- Use **`beam.WindowInto`** to define how your data is aggregated in EventTime
- Produce partial results
- How and when to handle late data

See
https://beam.apache.org/documentation/programming-guide/#triggers

Cross Language Transforms

- Found in beam/io/xlang/...
  - KafkaIO
  - BigqueryIO
  - JDBCIO
  - BeamSQL
  - DebeziumIO
- Automatic Java Expansion Service Startup

Learn more at
https://beam.apache.org/documentation/programming-guide/#multi-language-pipelines

Native Streaming

Author Unbounded Splittable DoFns in Go

- Self Checkpointing with Process Continuations
- Unbounded Restrictions for Splitting
- Watermark Estimation
- Bundle Finalization
- Custom Drain Trucation

Native Streaming

Writing a native Go streaming pipeline

Tuesday 16:15-16:40 CDT, Room 203
with Danny McCormick and Jack McCluskey
https://2022.beamsummit.org/sessions/native-go-pipeline/

Pipeline Results

```
func queryMetrics(pr beam.PipelineResult, ns, n string) metrics.QueryResults {
    return pr.Metrics().Query(func(r beam.MetricResult) bool {
        return r.Namespace() == ns && r.Name() == n
    })
}
```

Map Side Inputs

```
ProcessElement(..., lookup func(K) func(*V) bool,...){
...

vals := lookup(key)
var val V

for vals(&val) { ... }
```

Performance

- Side Input Cache
- Generic Registration
- Load Tests!

Cross Bundle Side Input Cache

Caches inputs on the SDK worker side for cross bundle access.

Enable* with the **`harnessopts`** package:

```
import “github.com/apache/beam/sdks/v2/go/pkg/beam/util/harnessopts”

harnessopts.SideInputCacheCapacity(keyCount)
```

***Requires runner state cache support**

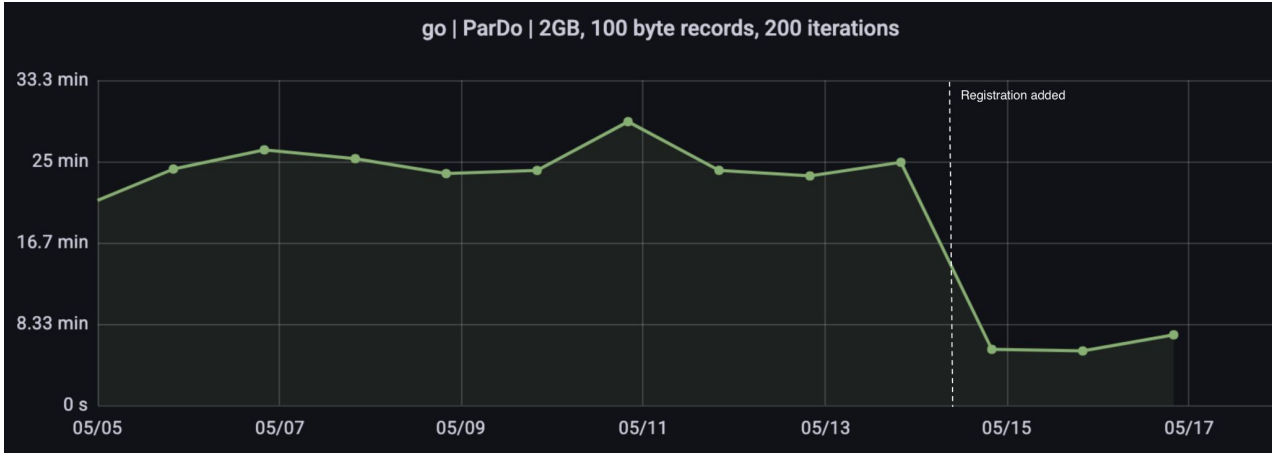
Generics for Performance

```
import “github.com/apache/beam/sdks/v2/go/pkg/beam/register”

func init() {
    register.DoFn2x1[KeyType,ValueType,ReturnType](&myDoFn{})
    register.Function4x0(simpleDoFn4x0)
}
```

Load Tests!

Covers various Batch ParDo, GBK, CoGBK, Combine, and Side Inputs patterns.



To see them select the suite and then Batch and Go at metrics.beam.apache.org

Dataflow Support

BEAM SUMMIT Austin, 2022

Thank you for contributing!

- @abacn
- @bamnet
- @ceocoder
- @damccorm
- @damondouglas
- @davidhinkes
- @gonzojive
- @htyleo
- @ibzib
- @ihji
- @inogueir
- @jcking
- @johnedmonds
- @johnjcasey
- @jrmccluskey
- @kamilwu
- @kw2542
- @lostluck
- @lukakalinovcic
- @milantracy
- @miracvbasaran
- @nguyennk92
- @riteshghorse
- @thempatel
- @tszerszen
- @y1chi
- @yichuan66
- @ymatzki
- @youngoli

BEAM SUMMIT

Summer 2022

What's next?

TODO(you?)

- State and Timers
  - GroupIntoBatches
- More
  - Native SDF IOs
  - Cross Language Wrappers
- Go SDK Expansion Service
  - Dynamic Row type
- Go
  - Faster!
  - Generics!

Related Talks

Oops, I wrote a Portable Beam Runner in Go

Tuesday 12:00-12:25 CDT, Room 203
with Robert Burke
https://2022.beamsummit.org/sessions/portable-go-beam-runner/

Writing a native Go streaming pipeline

Tuesday 16:15-16:40 CDT, Room 203
with Danny McCormick and Jack McCluskey
https://2022.beamsummit.org/sessions/native-go-pipeline/

State of the Go SDK 2022

Robert Burke
@lostluck

BEAM SUMMIT Austin, 2022

Reference Links

Experimental Exit

Go v2.40