Resource Hints

By: Valentyn Tymofieiev

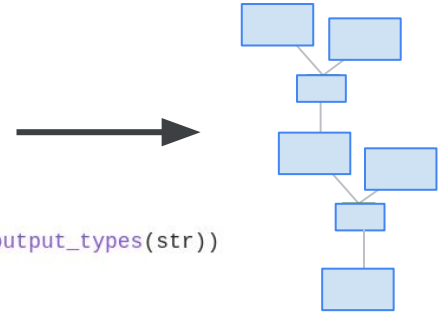
Software engineer @ Google, Apache Beam Committer

Beam portability framework

```
with beam.Pipeline(options=pipeline_options) as p:

  # Read the text file[pattern] into a PCollection.
  lines = p | 'Read' >> ReadFromText(known_args.input)

  counts = (
      lines
      | 'Split' >> (beam.ParDo(WordExtractingDoFn()).with_output_types(str))
      | 'PairWithOne' >> beam.Map(lambda x: (x, 1))
      | 'GroupAndSum' >> beam.CombinePerKey(sum))
```



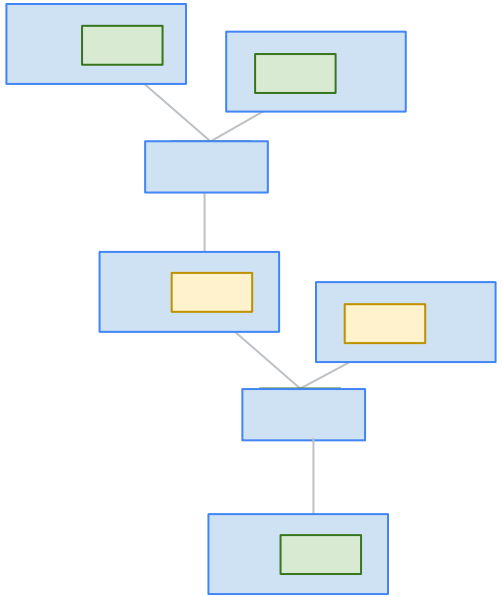
Write Pipeline

Translate

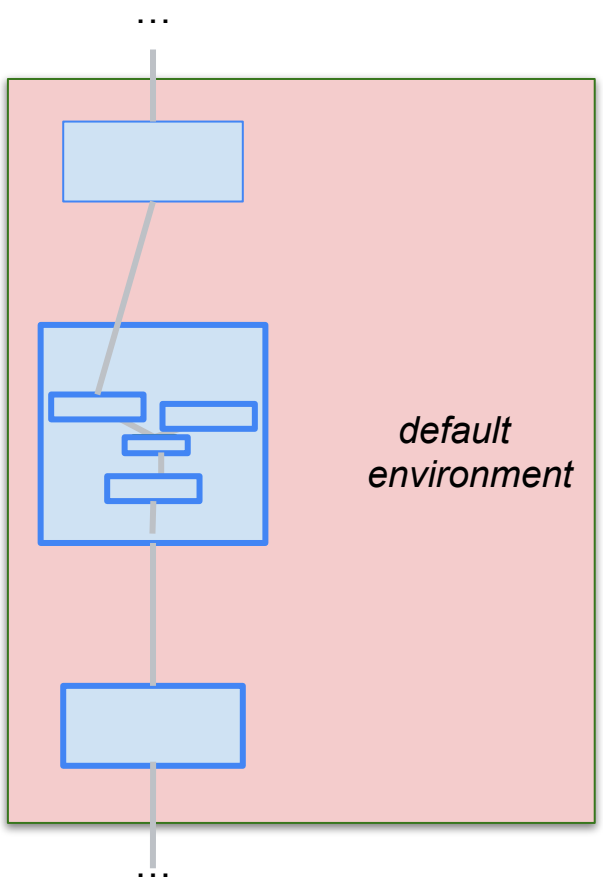
Execute

Translation

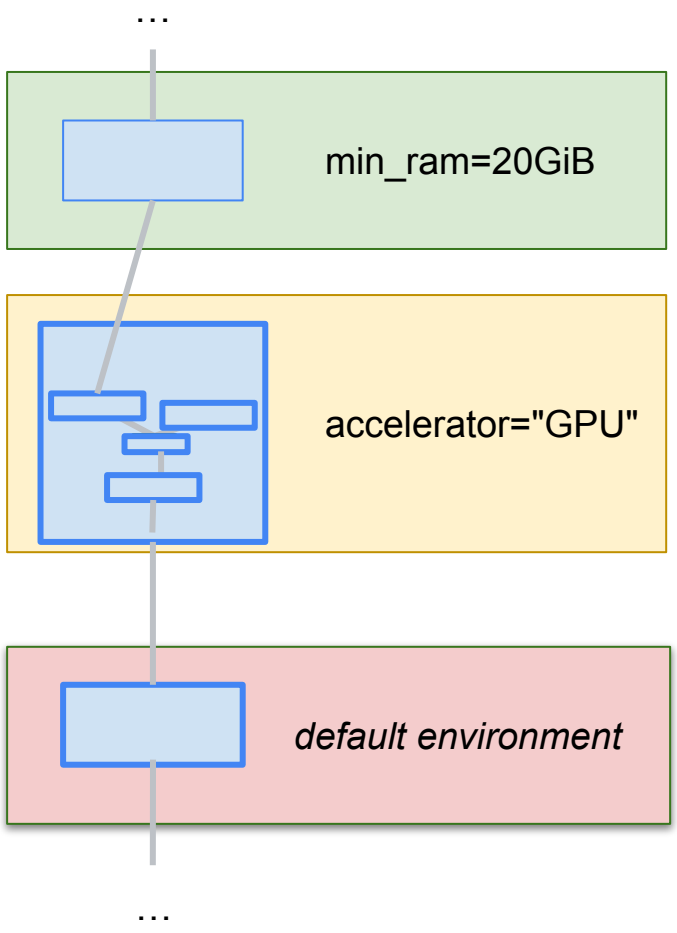
- The *Beam Runner API* defines a language-neutral, runner-independent pipeline representation
- Transformations specify **an environment** for their execution
- Resource hints on transforms prodive suggestions for the runners about the desired resources in the execution environment



```
p | "Read elements" >> ...
    | beam.ParDo(MyDoFn())
    | MyCompositePTransform()
    | beam.Map(...)
    | "Write elements" >> …
```



```
p | "Read elements" >> ...
    | beam.ParDo(MyDoFn()).with_resource_hints(min_ram="20GiB") |
    | MyCompositePTransform().with_resource_hints(accelerator="GPU")
    | beam.Map(...)
    | "Write elements" >> …
```



A peek into the pipeline representation proto

```
transforms {
  key: "ref_AppliedPTransform_ParDo-MyDoFn-_14"
  value {
    inputs { value: "ref_PCollection_PCollection_8" …  }
    outputs {...}
    unique_name: "ParDo(MyDoFn)"
    environment_id: "ref_Environment_environment_with_resource_hints_2"
  }
  …
}
environments {
  key: "ref_Environment_environment_with_resource_hints_2"
  value {
    urn: "beam:env:docker:v1"
    payload: "\n-gcr.io/cloud-dataflow/v1beta3/python37:2.38.0"
    capabilities: "beam:coder:custom_window:v1"
    capabilities: "beam:protocol:sibling_workers:v1"
     …
    resource_hints {
      key: "beam:resources:min_ram_bytes:v1"
      value: "21474836480"
    }
  }
}
```

Usages

- Indicate that transforms require specific resources
  - Memory, GPU, TPU or other specialized hardware
- Enables heterogeneous processing by runners
- Runners support is being added in new features
  - Example: https://cloud.google.com/dataflow/docs/guides/right-fitting

Resource hints contract

- Can be set in pipeline code
  - One some transforms or for entire pipeline (via Pipeline options)
- Common hints may be defined in Beam SDK
  - min_ram, accelerator
  - but also possible to register and supply a new hint at runtime
- They are hints:
  - Interpreted by runners
  - Unknown hints should be ignored
  - Expected syntax can be determined by runner maintainers
    - **Example: accelerator=**'type:nvidia-tesla-t4;count:1;install-nvidia-drivers
  - Runners can still decide fusion behavior for transforms with different environments

How can you help

- Users are encouraged to share what resource hints they would find useful, provide feedback to the runner maintainers
- Runner maintainers are encouraged to add support for resource hints and add new hints to the SDK.
- Learn more at: https://beam.apache.org/documentation/runtime/resource-hints/

Thank you!