OpenKasugai Controller (Extensions)

v1.1.0

Term definitions (1)

Terminology	Description
function chain	Definition of the features that make up the data processing flow and the connectivity relationships between the features. May be provided as a template. Also referred to as FC.
Function	A function defined in function chain, a data processing module.
data flow	You deploy function chain to a physical resource, and you deploy it. Also referred to as DF.
User	Users of this system. There are three types of users. (1) DF execution administrator (who requests DF deployment + who manages apps that receive input to DF and output from DF) (2) Operator of this system (Operator. Who provides DF to DF execution administrator) (3) Developers (people who develop features (such as functions) required by DF)
GW	Inlets and exits visible to DF execution administrator (Images like Kafka, RabbitMQ, and NetScaler. It could be a GW service in the cloud)
StartPoint (SP), EndPoint(EP)	data flow source and destination. Now it means Camera stream server. In the future, it will be expanded to include functions such as entry GW and exit GW, such as session trimming.
Toggle	data flow replacement process
scheduling condition	When deploying data flow, specify where to deploy (for example, directly (and/or uncandidly) or filter execution strategy) By selecting a defined scheduling condition during data flow deployment request, the deployment destination is determined according to those conditions
filter execution strategy	Specifying the filter to apply during scheduling (List of filters to be applied, their order, and how many of the top deployment destination candidates that pass the filter are to be used.)

Term definitions (2)

Terminology	Description
parent bs (parent bitstream)	The basic circuit of an FPGA. It can be used only after writing the following child bs
child bs (child bitstream)	Actual FPGA circuitry to be written on parent bs (e.g., resize circuitry)
child bs Write	any of the following three patterns of child bs writing
automatic writing	If an child bs unwritten FPGA is selected for deployment, dynamically write child bs before deploying DF function.
Manual Write	To write a child bs to an FPGA in a child bs unwritten state and make the controller recognize it as a new deployment candidate without stopping the controller during operation.
Overwrite	Writing child bs to child bs Written FPGAs
Reset	Generic term for the following two patterns of FPGA resets
Reset FPGA	Reverting to child bs Unwritten FPGA
Reset child bs	To restore the currently written child bs to its initial state immediately after writing. Initialize the connection information set during DF deployment.
Freeing FPGA Resources	To release resources of an FPGA consumed by DF deployment and return them to an unused state. This is an FPGA-side operation that has nothing to do with K8s resources or controllers.
bifurcation	A generic term for the following two patterns of branching. However, "copy branch" may be abbreviated depending on the context.
copy branch	copy one data to multiple destinations
conditional branch	Send a piece of data to a specific destination
Integration	Flow multiple processing results into one
Glue	A function that converts the connection type of a function for which I/O connection types are different (e.g., TCP→DMA) By deploying between functions with different connection types, you can deploy an DF that connects these functions (which cannot be connected).

Contents

1. Introduction
   - Value Proposition
   - Extensions List

2. DF branching into multiple functions
   - Copy Branch Processing Module
   - OpenKasugai Controller Enhancements

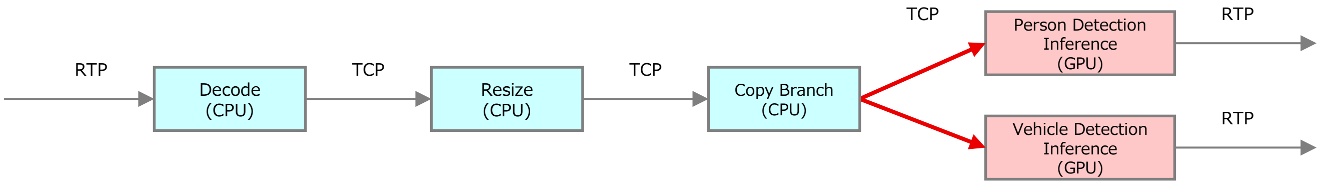
3. DF connecting functions of different connection types
   - Glue Processing Module
   - OpenKasugai Controller Enhancements

1. Introduction

Value proposition

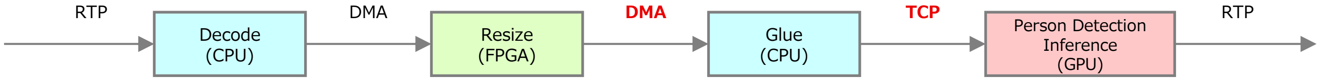
- Increase the variations of deployable DFs to enable the deployment of diverse data flows.
  - A copy branch processing module can be used to deploy data flows that branch into multiple functions along the way.
  - A connection type conversion module (Glue processing module) can be used to deploy data flows that connect functions with different connection types.

DF that branches into multiple functions



RTP
Decode (CPU)
TCP
Resize (CPU)
TCP
Copy Branch (CPU)
TCP
Person Detection Inference (GPU)
RTP
Vehicle Detection Inference (GPU)
RTP

DF connected between functions with different connection types



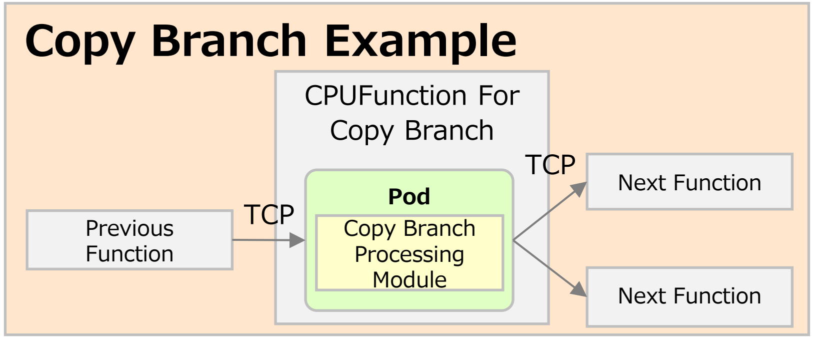
Extensions list

DF that can be built	Overview of Extensions
DF that branches into multiple functions	• Provide a copy branch processing module as a function. • Extend the controller to manage functions with multiple inputs and outputs. • Extend the controller to Define a DF with Branches.
DF connected between functions with different connection types	• Provide a Glue processing module as a function. • Extending the controller to manage functions with different input/output connection types.

2. DF that branches into multiple functions

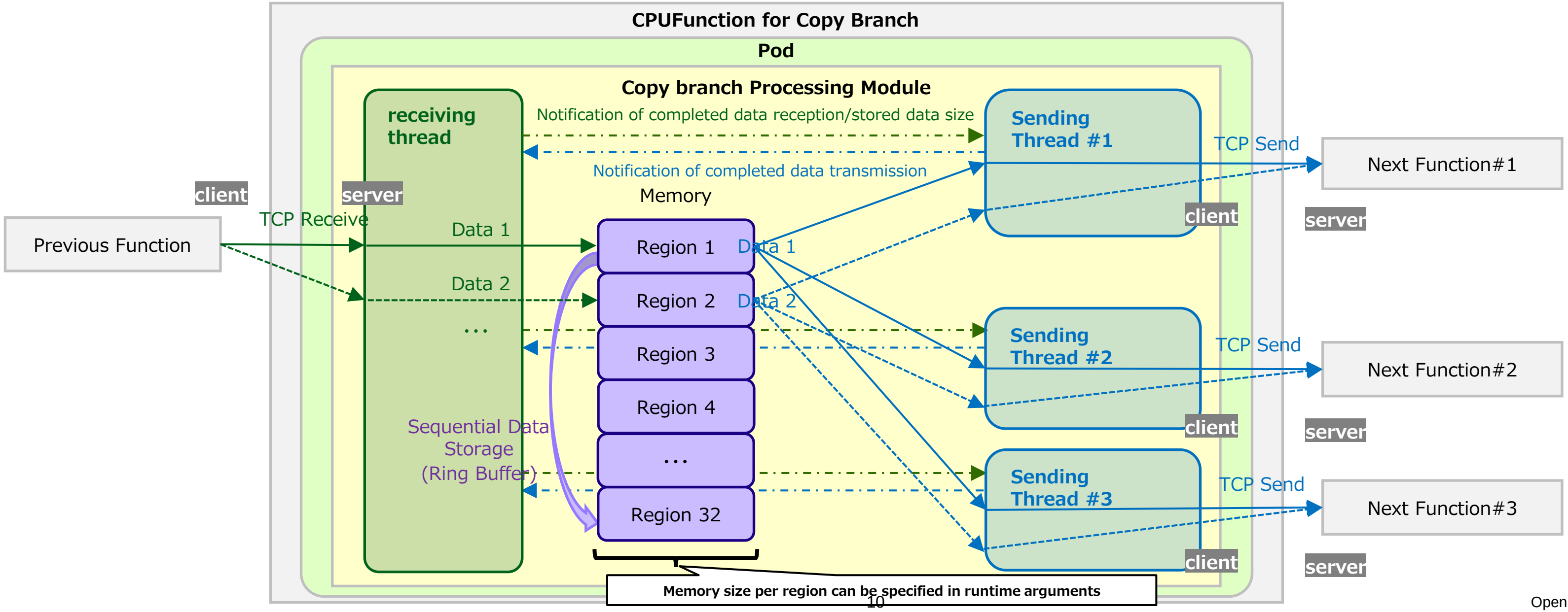
Copy branch processing module: Summary

- Processing for distributing and outputting input from a previous function to multiple next functions.
  - TCP input/TCP output
  - The number of branches can be set to any number.



Copy branch processing module : Block diagram

- The copy branch processing module consists of one receiving thread and sending threads corresponding to the number of branches.
    - The receiving/sending thread communicates using completion notifications prepared for each sending thread and each memory area.
    - When receiving data from the previous function and storing it in memory is completed, the receiving thread notifies the stored data size and the reception completion.
    - If data reception is completed, the sending thread transmits data corresponding to the stored data to the next function, and then notifies the transmission completion.



OpenKasugai controller enhancements: Managing functions with multiple inputs and outputs

- Extend the CR in the app catalog to manage the number of input and output ports for functions.
  - The maximum number of ports for input/output is described, and the DF definition determines the number of ports actually used.
  - Check the DF definition to ensure that the number of ports for each function in the DF definition does not exceed the maximum number of ports listed in the catalog.
  - Note)The "port" of a function is a cataloged concept,
    different from the "NIC" of a pod, etc.

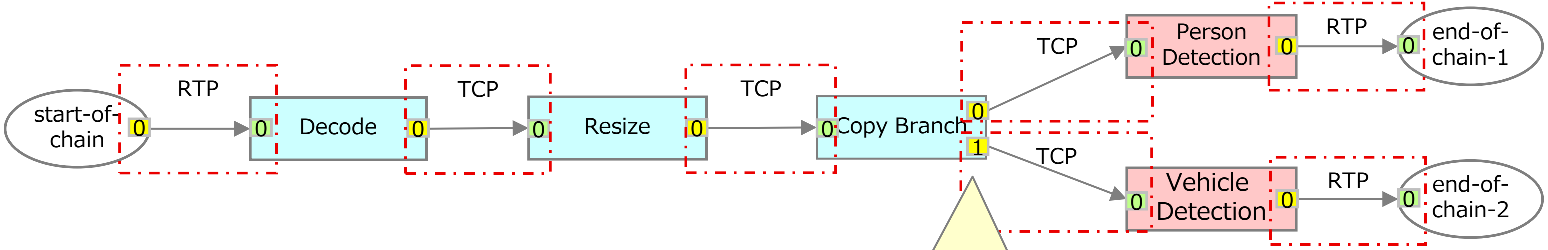
Catalog Example

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: funcinfo-copy-branch
  namespace: wbfunc-imgproc
data:
  deployableItems: '[
    {
      "name": "item1",
      "regionType": "cpu",
      "inputInterfaceType": "host100gether",
      "outputInterfaceType": "host100gether",
      "configName": "cpufunc-config-copy-branch",
      "specName": "spec1"
    }
  ]'
  spec: '[
    {
      "name": "spec1",
      "minCore": 1,
      "maxCore": 1,
      "maxDataFlowsBase": 1,
      "maxCapacityBase": 15,
      "maxInputNum": 1,
      "maxOutputNum":10
    }
  ]'
```

Describe the maximum number of connections

OpenKasugai controller enhancements:
Defining DF with branches

- Extend the data flow CR to represent a DF with multiple branching functions.
  - When branching, which port is connected is managed by number (number in square).
  - Multiple DF start and end points are also managed. (start-of-chain, end-of-chain-1,2)



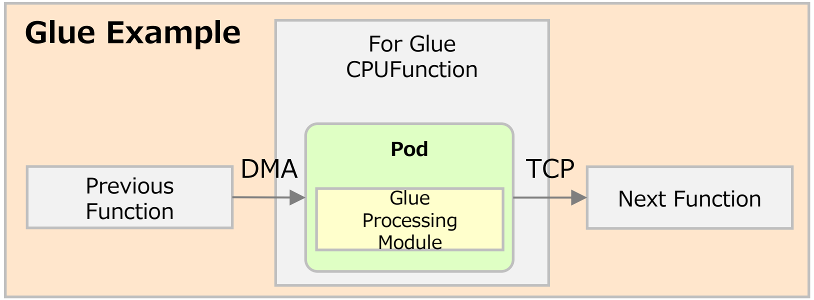
The person detection function is connected to the 0 output port of the copy branch function.

The vehicle detection function is connected to the 1 output port of the copy branch function.

3. DF connected between functions with different connection types

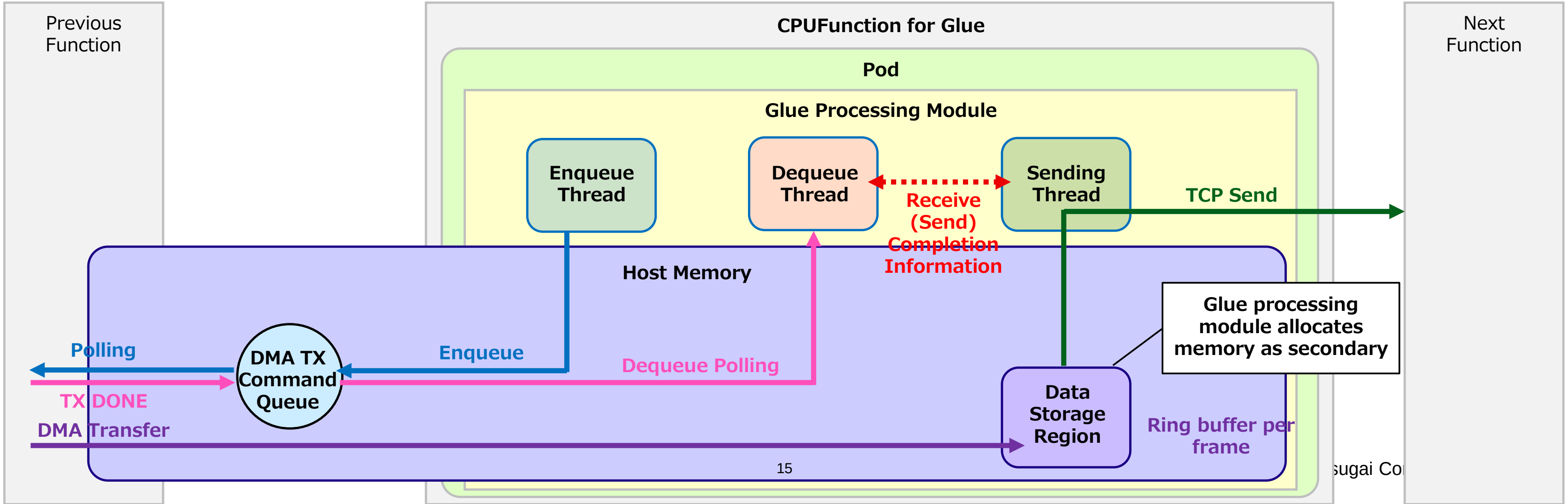
Glue processing module: Summary

- Process of connecting data transfers between different connection types.
  - DMA input/TCP output



Glue processing module: Block diagram

- The Glue processing module consists of an enqueue thread, a dequeue thread, and a send thread.
  - Make TX DMA requests to the DMA TX command queue on the enqueue thread and poll on the dequeue thread.
  - When the dequeue thread detects the completion DMA transfer of the previous function, it notifies the sending thread that it has completed receiving.
  - When the sending thread receives a completion notification, it sends the data to the next function via TCP and then notifies the dequeue thread of the completion.



OpenKasugai controller enhancements:
Managing functions with different Input/Output connection types

- Extending the CR of the app catalog to manage the connection type of the function input port and output port
  - As a check of DF definitions, check whether the connection type of each connection in the DF definition matches the connection types listed in the catalogs of the functions at both ends.

Catalog Example

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: funcinfo-glue-fdma-to-tcp
  namespace: wbfunc-imgproc
data:
  deployableItems: '[
    {
      "name": "item1",
      "regionType": "cpu",
      "inputInterfaceType": "mem",
      "outputInterfaceType": "host100gether",
      "configName": "cpufunc-config-glue-fdma-to-tcp",
      "specName": "spec1"
    }
  ]'
  spec: '[
    {
      "name": "spec1",
      "minCore": 1,
      "maxCore": 1,
      "maxDataFlowsBase": 1,
      "maxCapacityBase": 15,
      "maxInputNum": 1,
      "maxOutputNum":1
    }
  ]'
```

Indicates the input/output connection type.