OpenKasugai Controller

v1.0.0

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
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.
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

- Introduction
  - Objectives and Approaches
  - Value proposition aiming for
- Overview of the OpenKasugai controller
  - Target Hardware Configuration
  - Deployable data flow
  - Basic concept
- Overall configuration of the OpenKasugai controller
  - Overall configuration of the OpenKasugai controller
  - Three functions of the OpenKasugai controller
  - Component List
- Functional description of the OpenKasugai controller
  - Scheduling function
  - Basic deployment functions
  - Infrastructure information collection and management function
- Overall flow of operations

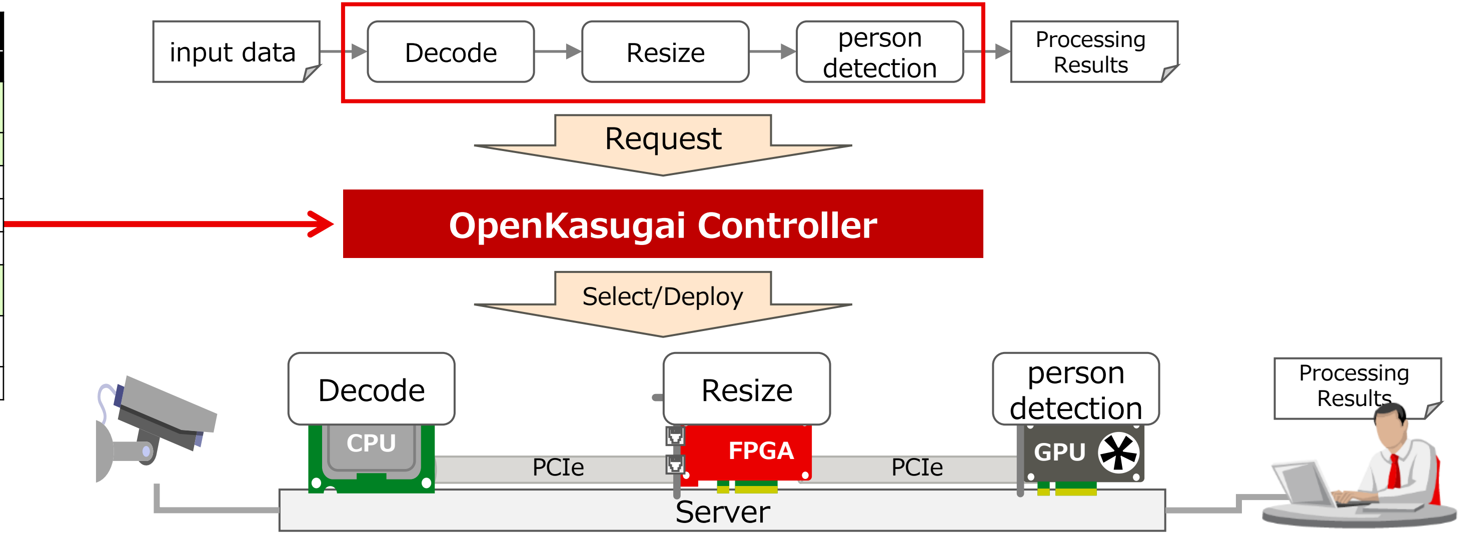
1. Introduction

Objectives and approaches

- Facilitate data processing using accelerators
  - Catalog proven accelerator processing as abstract functional blocks
  - Leverage accelerators to connect cataloged processes to achieve required data processing
  - Select the appropriate accelerator/connection type and deploy/connect

processing catalog

Function	Processing	Accelerator
video preprocessing	Decode	CPU
	Resize	FPGA
		CPU
	smoothing	FPGA
	Grayscale	GPU
inference processing	person detection	GPU
	object detection	GPU
…	…	…



Value proposition aiming for

- Data flow can be automatically deployed
  - To avoid communication failure and performance degradation due to resource capacity excess even if a user is unaware of it.
    - = Manage resources where DF is deployed
  - Deploy an DF with better performance and power consumption without the user's awareness
    - = Choose better devices and routes for performance and power consumption
  - Each DF can be deployed in a different scheduling strategy (if the user wants)
    - = Prepare various scheduling condition in advance. Choose a destination according to the criteria you select in DF definition

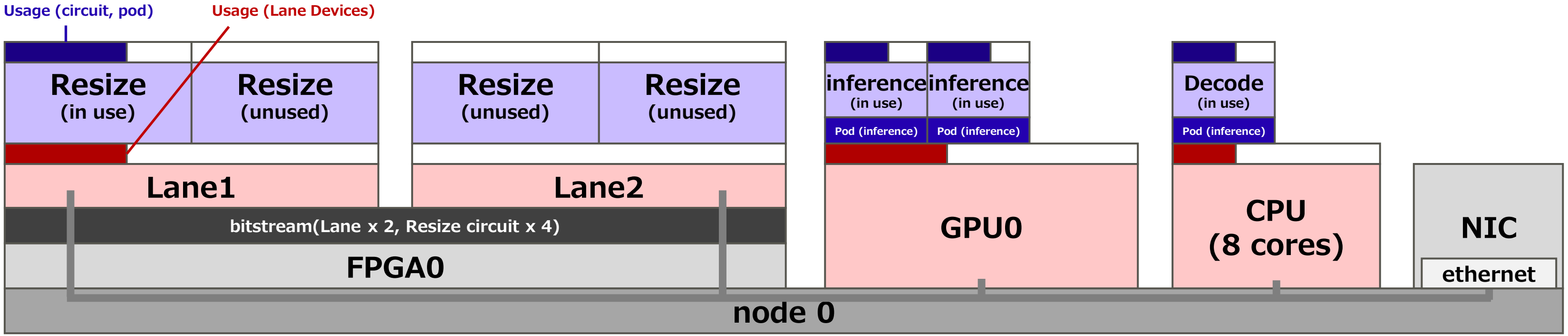
- Reduce the amount of manual user changes when changing the environment
  - Automatic collection and resource registration of environment-dependent information from the infrastructure to avoid manual operation by users
  - Reduce user workload by creating common information that does not depend on the environment or DF and performing automatic completion according to the environment or DF

2. Overview of the OpenKasugai controller

Target hardware configuration

Device	Hardware Configuration	data flow Deployment Configuration
FPGA	• Split an FPGA into two Lane (regions) to run multiple circuits on Lane • Multiple Lane and multiple circuits are written together due to hardware limitations.	• Multiple data flow are assigned to a circuit • Select deployment locations per Lane or per circuit on Lane
GPU	• Deploy multiple pods on a single GPU	• 1 data flow per pod • Scheduling selects deployment locations per GPU
CPU	• Collectively manage all CPU cores on the server	• 1 data flow per pod • Scheduling selects all CPU cores at once • Let the K8s decide which core to actually use.

- Reference) Function Definition
  - When registering a function, be aware of the deployment destination area (Lane/GPU model/CPU), and register the area to which it can be deployed and the required resource amount and performance.
  - The I/F (IO) type (TCP/DMA) to be used for each function is fixed, and which one to use is also registered. Do not select a destination where that IO is not available



Deployable data flow

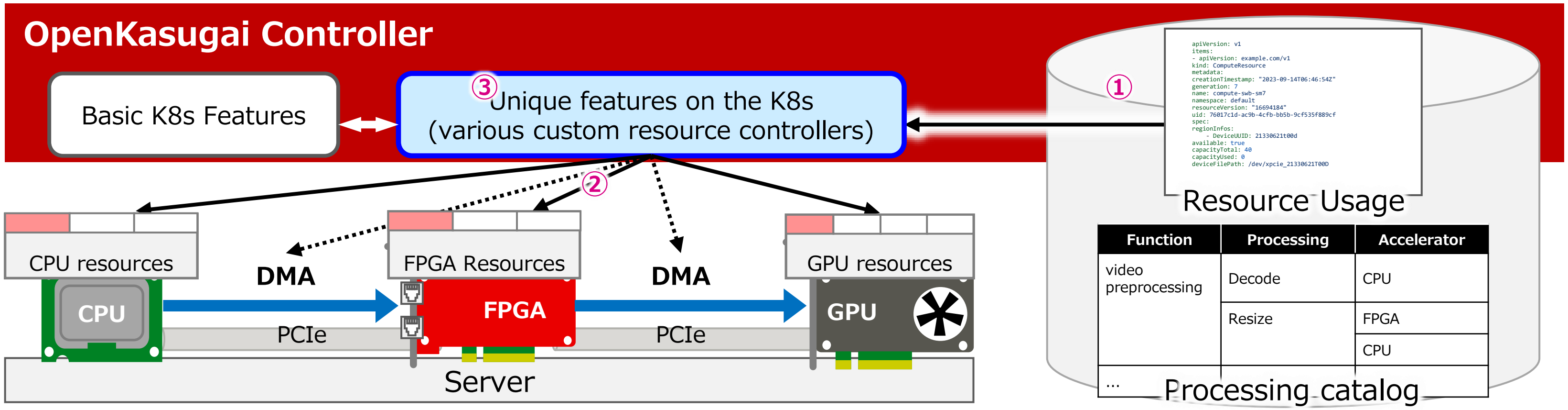
- You can specify and deploy the functions you want to use. Assume the following functions
  - decoding: CPU version present
  - Resize: FPGA version and CPU version. Input/output size can be set according to the inference in the later stage.
  - Inference: There are two GPU versions of advanced inference (1280x1280 with A100) and lightweight inference (416x416 with T4). There are two types (person detection and vehicle detection)
    (4 types in total)
- TCP or DMA can be selected as the connection type between functions.
- You can deploy each function without specifying where to deploy it.
  - Optimal destination selection across DF (scheduling)
    - Select the appropriate deployment destination combination based on connection type and free space
    - When deploying functions that support multiple connection types, scheduler appropriately selects the connection type itself based on catalog information
    - You can specify the name of scheduler condition you want to apply in DF definition. If specified, can be deployed to the specified destination
- Data flow can request multiple deployments in a row

processing catalog

Function	Processing	Accelerator
video preprocessing	Decode	CPU
	Resize	FPGA
		CPU
inference processing	People Detection (Advanced)	GPU
	People Detection (Lightweight)	GPU
	Vehicle Detection (Altitude)	GPU
	Vehicle Detection (Light)	GPU

Basic concept (1)

1. Manage and utilize accelerator resource usage
   - Manage accelerator resource usage. Select the appropriate accelerator type and connection type according to the usage conditions
2. Software controls processing deployment to and connections between accelerators
   - Perform the operation according to the selected accelerator type or connection type.
3. Leveraging Kubernetes (K8s) extensibility mechanisms (custom resource)
   - Manage data processing pipelines, resource usage, accelerators and inter-accelerator connections with custom resource



```
apiVersion: v1
items:
- apiVersion: example.com/v1
kind: ComputeResource
metadata:
creationTimestamp: "2023-09-14T06:46:54Z"
generation: 7
name: compute-swb-sm7
namespace: default
resourceVersion: "16694184"
uid: 76017c1d-ac9b-4cfb-bb5b-9cf535f889cf
spec:
regionInfos:
    - DeviceUUID: 21330621t00d
available: true
capacityTotal: 40
capacityUsed: 0
deviceFilePath: /dev/xpcie_21330621T00D
```

Function	Processing	Accelerator
video preprocessing	Decode	CPU
	Resize	FPGA
		CPU
…		

Basic concept (2)

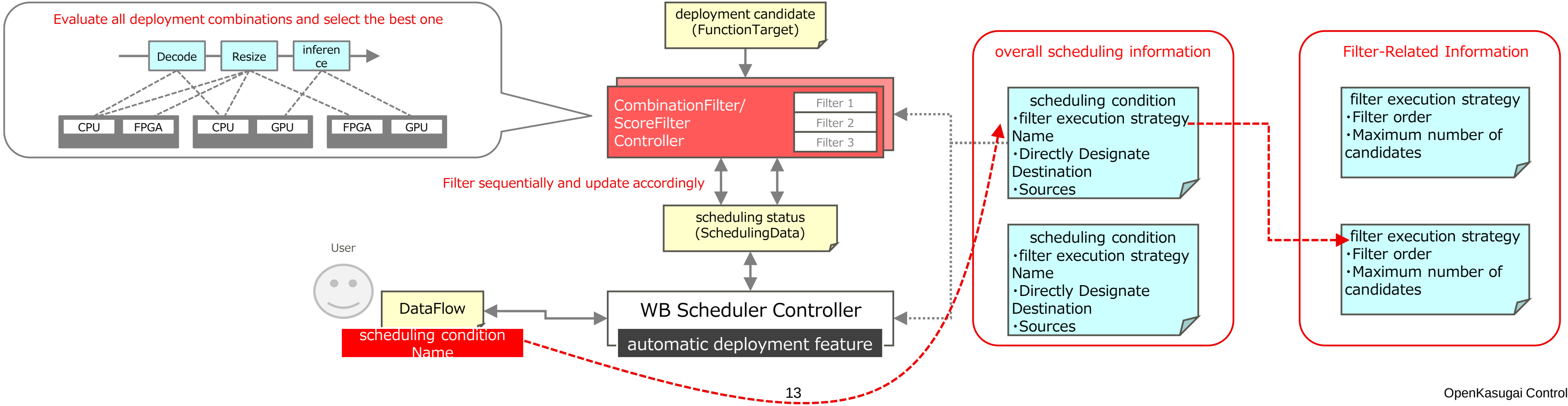
- Scheduling in declared value base
  - Managing the Maximum Processing Performance (fps) of Nodes and Accelerators
  - The estimated load (fps) is included in DF information, and scheduler decides whether to install it.
    - Check the availability (difference between the maximum processing performance and the current load) of the accelerator at the deployment destination to check whether the expected load can be accommodated.

- Manage infrastructure usage in declared value base
  - Deploying DF increases the current load on the destination accelerator by the expected load
  - When DF is deleted, each current load is reduced by the assumed load.

- Choose where to deploy from the available node accelerators to the node accelerator you are using as much as possible
  - More detailed or alternative destination selection methods can also be specified in scheduling condition on the next page

Basic concept (3)

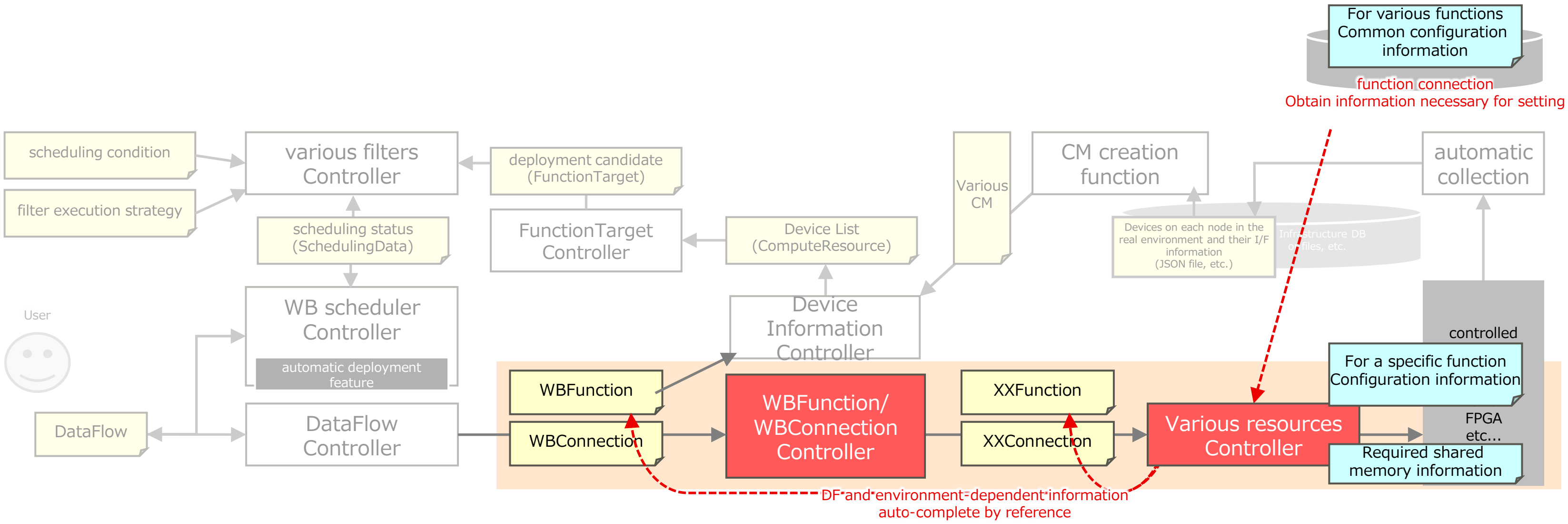
- DF scheduler decides where to deploy along scheduling condition selected by the user during deployment request
  - scheduling condition example
    - Filtering/scoring plug-in combination to apply
    - Number of destination candidates after filtering/scoring is applied (how many top candidates remain)
    - Directly Designate Destination Nodes and Devices (Acceptable or excluded conditions)
    - Specify the information source to refer to when selecting a deployment destination (when information for multiple deployment destination candidates is managed)
  - scheduling condition can be used across DF by making it a separate resource from DF definition
    - In the future, we will provide various types of scheduling condition, such as power-oriented and performance-oriented, for users to choose from.



Evaluate all deployment combinations and select the best one
Decode
Resize
inferen ce
CPU
FPGA
CPU
GPU
FPGA
GPU
deployment candidate (FunctionTarget)
CombinationFilter/ ScoreFilter Controller
Filter 1
Filter 2
Filter 3
Filter sequentially and update accordingly
scheduling status (SchedulingData)
User
DataFlow
scheduling condition Name
WB Scheduler Controller
automatic deployment feature
overall scheduling information
scheduling condition
・filter execution strategy Name
・Directly Designate Destination
・Sources
scheduling condition
・filter execution strategy Name
・Directly Designate Destination
・Sources
Filter-Related Information
filter execution strategy
・Filter order
・Maximum number of candidates
filter execution strategy
・Filter order
・Maximum number of candidates

Basic concept (4)

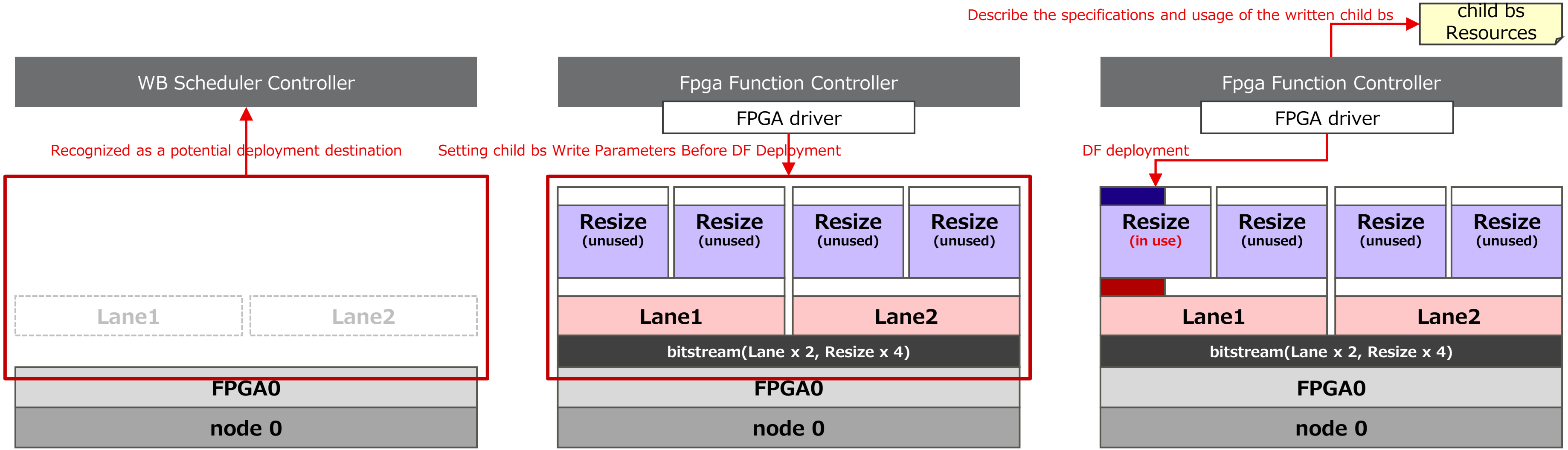
- Automatic completion of information required for the deployment of various functions (FPGA, GPU, and CPU functions)
  - By making the information to be prepared in advance independent of environment and DF, it is possible to use it in common.
    - Common configuration information for various functions
  - Auto-complete environment and DF dependent information required for actual deployment
    - Information about the devices to which the function is deployed before and after the function, information about DF to which the function belongs, etc.
    - Automatic completion of shared memory information required for PCIe connection from DF information and function config files (shared memory name and size)



For various functions Common configuration information
function connection
Obtain information necessary for setting
scheduling condition
filter execution strategy
various filters Controller
deployment candidate (FunctionTarget)
scheduling status (SchedulingData)
FunctionTarget Controller
Device List (ComputeResource)
Various CM
CM creation function
Devices on each node in the real environment and their I/F information (JSON file, etc.)
Infrastructure DB or files, etc.
automatic collection
User
WB scheduler Controller
automatic deployment feature
Device Information Controller
controlled
DataFlow
DataFlow Controller
WBFunction
WBConnection
WBFunction/ WBConnection Controller
XXFunction
XXConnection
Various resources Controller
For a specific function Configuration information
FPGA etc...
Required shared memory information
DF and environment-dependent information
auto-complete by reference

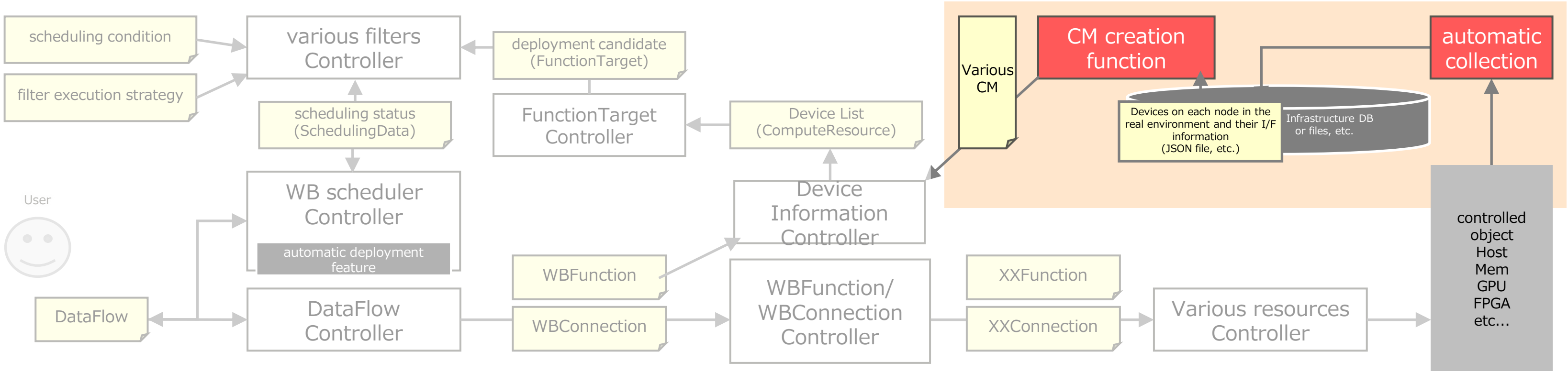
Basic concept (5)

- Dynamically writes child bs as needed during operation without having to manually write child bs to FPGA beforehand
  - child bs recognizes unwritten FPGAs as potential destinations and scheduler selects destinations from child bs written/unwritten FPGAs
  - If an child bs unwritten FPGA is selected for deployment, dynamically write child bs and set parameter values during deployment
    - Multiple Lane on the same FPGA must be bulk written due to hardware limitations, creating the same circuit resource for all Lane when dynamically writing child bs
    - Set parameter values for the requested function for all Lane
  - For child bs writes, create and manage a k8s resource that contains the written child bs information.



Basic concept (6)

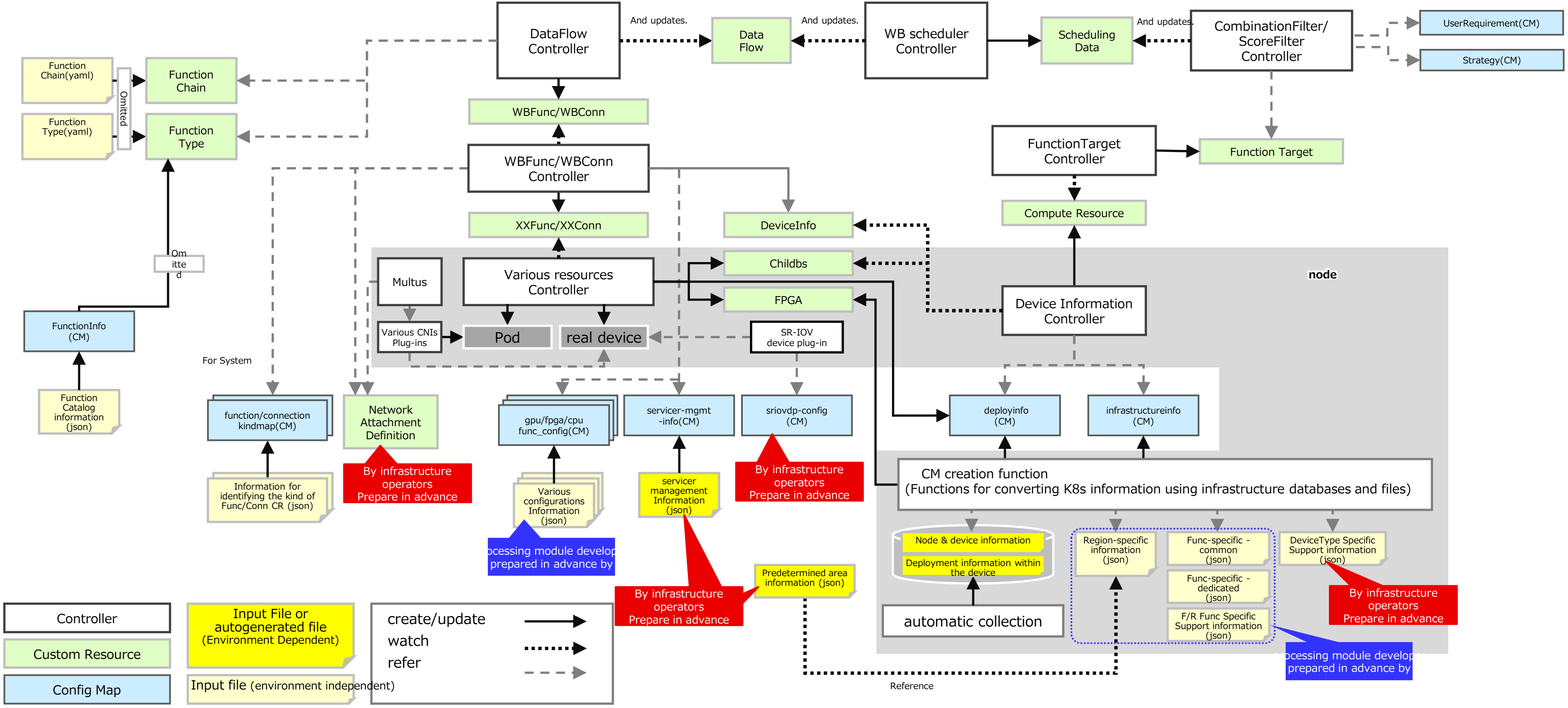
- Automatically collects input information previously manually and creates K8s resources for various controllers
  - Make certain parts of information that are independent of the environment common and usable
    - Catalog information for accelerators, circuits, and Pod
  - Automatically collect some environment-dependent information
    - Accelerator list (FPGA/GPU/CPU) on the node. Configuration of the region written on the FPGA
  - All of the above information is stored in the infrastructure database and automatically converted into K8s resources.



scheduling condition
filter execution strategy
various filters Controller
deployment candidate (FunctionTarget)
scheduling status (SchedulingData)
FunctionTarget Controller
Device List (ComputeResource)
Various CM
CM creation function
automatic collection
Devices on each node in the real environment and their I/F information (JSON file, etc.)
Infrastructure DB or files, etc.
User
WB scheduler Controller
automatic deployment feature
Device Information Controller
controlled object
Host
Mem
GPU
FPGA
etc...
DataFlow
DataFlow Controller
WBFunction
WBConnection
WBFunction/ WBConnection Controller
XXFunction
XXConnection
Various resources Controller

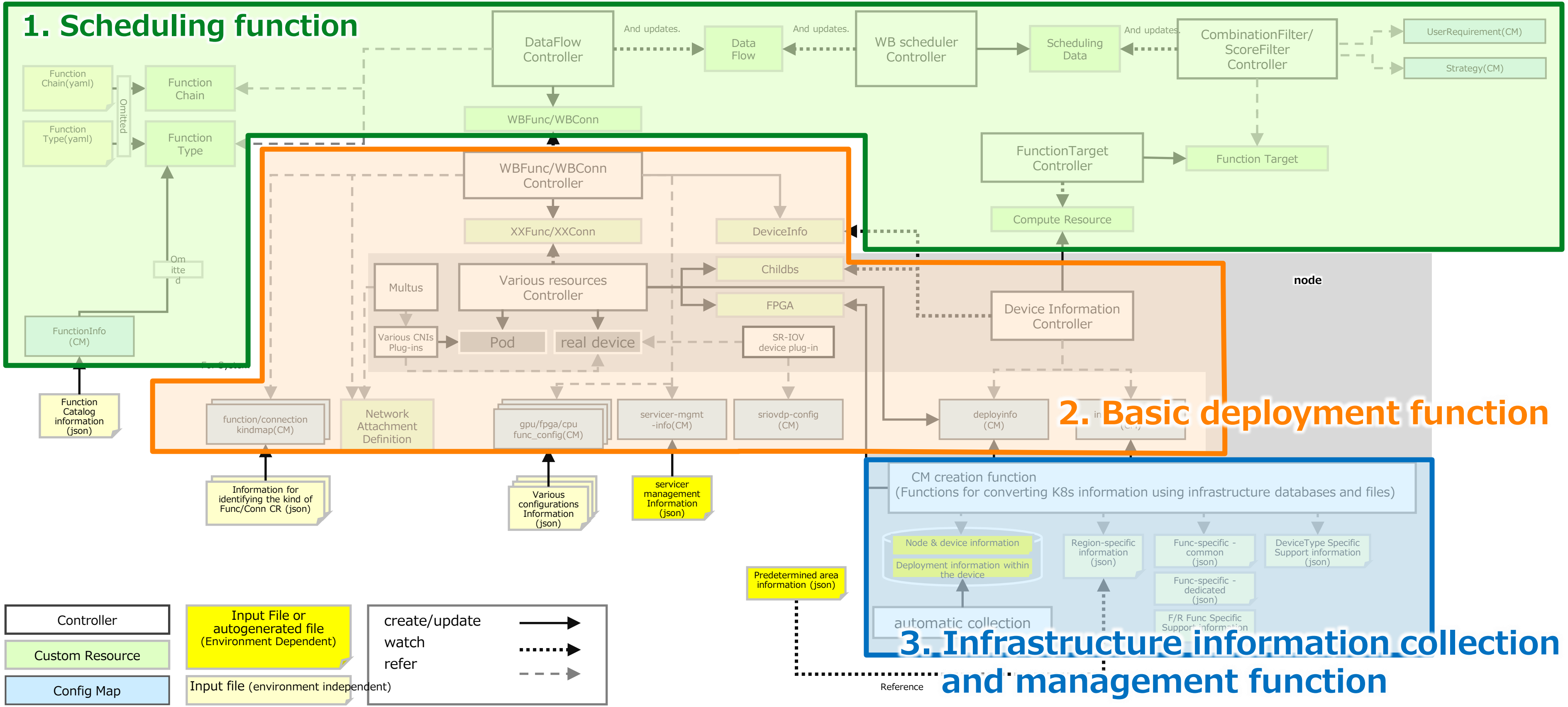
3. Overall configuration of the OpenKasugai controller

Overall configuration of the OpenKasugai controller


DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
FunctionTarget Controller
Function Target
Compute Resource
Om itte d
FunctionInfo (CM)
Function Catalog information (json)
Multus
Various resources Controller
Childbs
FPGA
node
Device Information Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
For System
function/connection kindmap(CM)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
infrastructureinfo (CM)
Information for identifying the kind of Func/Conn CR (json)
By infrastructure operators Prepare in advance
Various configurations Information (json)
servicer management Information (json)
By infrastructure operators Prepare in advance
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
ocessing module develop prepared in advance by
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
DeviceType Specific Support information (json)
By infrastructure operators Prepare in advance
Predetermined area information (json)
By infrastructure operators Prepare in advance
automatic collection
ocessing module develop prepared in advance by
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

See the following pages for descriptions of controllers and resources.

Three functions of the OpenKasugai controller


1. Scheduling function
DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Chain
Omitted
Function Type(yaml)
Function Type
WBFunc/WBConn
FunctionTarget Controller
Function Target
Compute Resource
Omitted
FunctionInfo (CM)
For System
Function Catalog information (json)
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Device Information Controller
node
function/connection kindmap(CM)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
in (CM)
2. Basic deployment function
Information for identifying the kind of Func/Conn CR (json)
Various configurations Information (json)
servicer management Information (json)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information
DeviceType Specific Support information (json)
automatic collection
Predetermined area information (json)
Reference
3. Infrastructure information collection and management function
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

See the following pages for descriptions of controllers and resources.

Three functions of the OpenKasugai controller

Function name	Description
Scheduling function	• Automatically determines where each data flow component will be deployed in response to data flow deployment requests, taking into account scheduling condition selected by the user and current usage • Demand that basic deployment functions deploy each component of data flow (WBFunction/WBConnection) • You can register FC templates and associated functions and connections.
Basic deployment function	• In response to deployment requests for each data flow component, use various resource controllers to actually deploy and configure the components. • Dynamically writes child bs if child bs is not written to the destination FPGA • The controller automatically completes the detailed parameters required for the various FPGA circuit and Pod. For example, channel number, IP address, port number • The results of deployment are provided to scheduling function as the usage status of various resources.
Infrastructure information collection and management function	• Provide automatic collection of infrastructure information and automatic creation of K8s resources • Automate the collection and use of environment-dependent information such as infrastructure configuration information and written area information on k8s • Automatic completion of DF dependent information (different information for each DF)

Component list:
Controller overview (1)

Item		Summary	Related Features
CRC	FunctionType Controller	Create a FunctionType Resource from a User's Registration Information (It is omitted in the figure. The results of joint research with NTT are used almost as is.)	scheduling function
	FunctionChain Controller	Create a FunctionChain Resource from a User's Registration Information (It is omitted in the figure. The results of joint research with NTT are used almost as is.)	scheduling function
	DataFlow Controller	deploy data flow at data flow's request	scheduling function
	FunctionTarget Controller	Create target information (GPU/FPGA/CPU) used by scheduler from ComputeResource resources (device information)	scheduling function
	WB Scheduler Controller	Automatically select an appropriate destination when data flow requests it, if the user has not specified a destination directly Check the amount of free space on the device and, if more than one candidate exists, use the score calculation to select the appropriate candidate.	scheduling function
	CombinationFilter Controller	Update SchedulingData by browsing SchedulingData and filtering the combination of deployment candidates based on function type and resource status.	scheduling function
	ScoreFilter Controller	Update SchedulingData by browsing SchedulingData and scoring candidate combinations based on device resource availability	scheduling function
	WBFunction Controller	Create various functional resources (GPUFunction/FPGAFunction/CPUFunction) as entities from WBFunction resources	basic deployment functions
	WBConnection Controller	Create various underlying connection resources (EthernetConnection/PCIeConnection) from WBConnection resources	basic deployment functions
	Various resource controllers	GPUFunction, FPGAFUnction, and CPUFunction are deployed. Also, set the connection between devices according to EthernetConnection and PCIeConnection.	basic deployment functions
	Device Information Controller	do the following ・Create an Initial ComputeResource Resource (Device Information) ・Update ComputeResource based on deployment status	basic deployment functions

Component list:

Controller overview (2)

	Item	Summary	Related Features
Others	Multus	Serves as a meta plugin to call various CNI plugins	basic deployment functions
	Various CNI plug-ins	Configure NIC and NW settings for Pod. Uses Calico (existing) for Pod K8s overlay NW and SR-IOV CNI (new) for 2nd NIC NW. Also utilizes static (new) as a CNI IPAM plug-in for 2nd NICs	basic deployment functions
	SR-IOV device plug-in	Recognize the VF of each node's NIC and expose it as a resource available from the K8s Pod	basic deployment functions
	CM creation function	Automatically create and register necessary ConfigMaps based on information in the infrastructure database (JSON file) # 1	Infrastructure information collection and management function
	Automatic collection	Collects information from the infrastructure and stores it in an infrastructure database (JSON file) # 1	Infrastructure information collection and management function

※1. Currently implemented as a single tool (InfoCollector)

Component list:
Resource overview (1)

	Name	Summary	Related Features	Manual/Automatic
CR	DataFlow	custom resource with information about the configuration (function and connection specifications) and requirements (expected load) of data flow you want to deploy	scheduling function	Created by DF execution administrator or operator
	FunctionChain	custom resource to represent the composition of data flow	scheduling function	Created by Operator or Created by Developer
	FunctionType	custom resource representing a Function available in function chain	scheduling function	Created by Operator or Created by Developer
	SchedulingData	custom resource with information on DataFlow's scheduling status. During scheduling, information such as the current pool of deployment candidates, the order in which filters are applied, and applied filters is managed to share status between controllers	scheduling function	Auto
	ComputeResource	custom resource containing information about the hard configuration and capacity management of each node. Indicates CPU, GPU, and FPGA information on the node. Also has information about device capacity management	scheduling function	Auto
	FunctionTarget	custom resource that has information about the candidate locations for functions built from ComputeResource	scheduling function	Auto
	WBFunction	custom resource for the Function to deploy. Converted to one of the following FJ versions of XXXFunction during deployment	scheduling function	Auto
	WBConnection	custom resource for the Connection to deploy. During deployment, it is converted to one of the following FJ versions of XXXConnection:	scheduling function	Auto
	GPUFunction	FJ version of custom resource with information about Functions deployed on GPUs (deployed on Pod with GPU)	basic deployment functions	Auto
	FPGAFunction	custom resource with information about the Function to be deployed on FPGA for FJ (phase3)	basic deployment functions	Auto
	CPUFunction	FJ version of custom resource with information about functions deployed on the CPU (deployed on Pod)	basic deployment functions	Auto
	EthernetConnection	FJ version of custom resource with information about Ethernet connections	basic deployment functions	Auto
	PCIeConnection	FJ version of custom resource with information about PCIe connections over shared memory	basic deployment functions	Auto
	DeviceInfo	custom resource with information exchanged between WBFunction and DeviceInfo controllers	basic deployment functions	Auto
	FPGA	custom resource with information about FPGA devices. Contains information on usage status and status management for writing and setting	basic deployment functions	Auto
	Childbs	custom resource with information about child bs written into the FPGA. Contains information about configured parameter values and DF deployment status	basic deployment functions	Auto
	NetworkAttachmentDefinition	custom resource with configuration information for Pod 2nd NIC referenced by Multus. Create one for every 100 GNICs on each node	basic deployment functions	Created by the operator

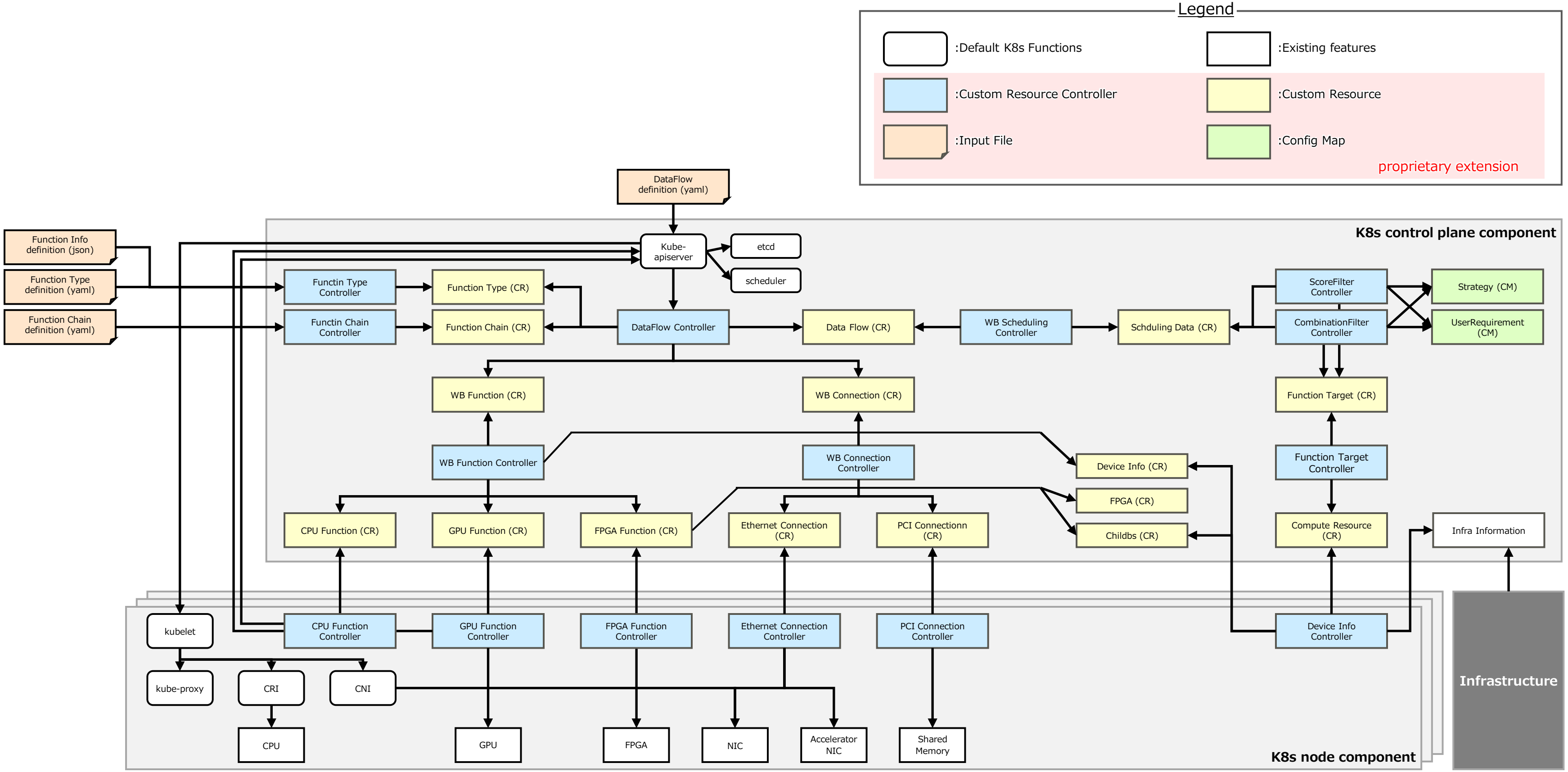
Component list:
Resource overview (2)

Name		Summary	Related Features	Manual/Automatic
CM	UserRequirement	Information that specifies Strategy configuration map used for DataFlow scheduling and filtering conditions for function/connection deployment destinations	scheduling function	Created by DF execution administrator or operator
	Strategy	Information specifying the execution strategy of the filter in DataFlow scheduling	scheduling function	Created by DF execution administrator or operator
	FunctionInfo	information equivalent to function catalog information	scheduling function	Created by Operator or Created by Developer
	infrastructureinfo	Information to define the hard configuration of each node. Define deployment space information for each device (GPU, FPGA, CPU)	basic deployment functions	automatic collection
	deployinfo	Information to define the deployable space provided on each node. Define deployment area information for each device (GPU, FPGA, CPU). It also has information about the maximum capacity of the device	basic deployment functions	automatic collection
	fpgacatalogmap	Information on FPGAs to be issued to each FPGAFunction	basic deployment functions	automatic collection
	functionkindmap	Information to identify which Function CR (GPUFunction/FPGAFunction/CPUFunction) to convert from WBFunction	basic deployment functions	Created by the operator
	connectionkindmap	Information to identify which Connection CR (EthernetConnection/PCIeConnection) to convert from WBConnection	basic deployment functions	Created by the operator
	gpufunc_config	ConfigMap containing configuration information for GPUFunction. Created per GPUFunction	basic deployment functions	Created by the operator
	fpgafunc_config	ConfigMap containing configuration information for FPGAFunction. Created per FPGAFunction	basic deployment functions	Created by the operator
	cpufunc_config	ConfigMap containing configuration information for CPUFunction. Created per CPUFunction	basic deployment functions	Created by the operator
	fpgalist-ph3	ConfigMap containing the information (Lane information and network information) to be configured for each FPGA at EthernetConnection startup	basic deployment functions	Created by the operator
	sriovdp-config	ConfigMap containing information about devices managed by the SR-IOV device plug-in	basic deployment functions	Created by the operator

Component list:
Resource overview (3)

	Name	Related CM	Summary	Related Features	Manual/Automatic
Input and auto-generated file (json)	Function catalog information	FunctionInfo	Basic information about various functions (Deployable accelerators, supported connection types, capacity, etc.)	scheduling function	Created by the operator
	Various configuration information	XXXfunc_config	Configurations that can be used in common by all DF functions. The controller complements this with the specific settings for each DF, resulting in the actual configuration information. (Since it is no longer necessary to create a configuration for each DF, a common configuration is assumed to be created by the developer.)	basic deployment functions	Created by Developer
	FPGA initial setup information	fpgalist-ph3	Information used to configure the FPGA (child bs). Currently, information set in the communication section of the FPGA, etc. (MAC, IP, Subnet, Gateway, etc.)	basic deployment functions	Created by the operator
	Func/Conn CR Kind identification correspondence information	functionkindmap, connectionkindmap	Mapping information prepared by the system that describes which function type and connection type in basic deployment functions the WB function and WB connection created by scheduling function correspond to	basic deployment functions	Fixed by the controller
	Node & device information	infrastructureinfo	Information about the list of nodes in the actual infrastructure and the list of devices installed in the nodes that are automatically collected	Infrastructure information collection and management function	automatic collection
	Device type identification correspondence information		Mapping information for converting automatically acquired device model names to DeviceType	Infrastructure information collection and management function	Fixed by the controller
	In-Device Deployment Information	deployinfo	Area information (FPGA child bs) written to each device that is automatically collected, and information about functions deployed in each area	Infrastructure information collection and management function	automatic collection
	Information for identifying F/R Func		Mapping information for determining whether an FPGA resize function is for advanced or lightweight inference	Infrastructure information collection and management function	Fixed by the controller
	Region-specific information		Information specific to the circuit (child bs) that writes the region. For example, the number and size of regions. Keep the same information for GPUs	Infrastructure information collection and management function	Created by the operator
	Func-specific - common		Common information (ID, Name, number of available WBFunc, capacity) among information fixed by circuit or container	Infrastructure information collection and management function	Created by Developer
	Func-specific - dedicated	fpgacatalogmap	Dedicated information for decoding and resizing among information fixed by circuits and containers. Prepare for each type of circuit	Infrastructure information collection and management function	Created by Developer

(Reference) Relationship between OpenKasugai controller and K8s function



4. Functional description of the OpenKasugai controller

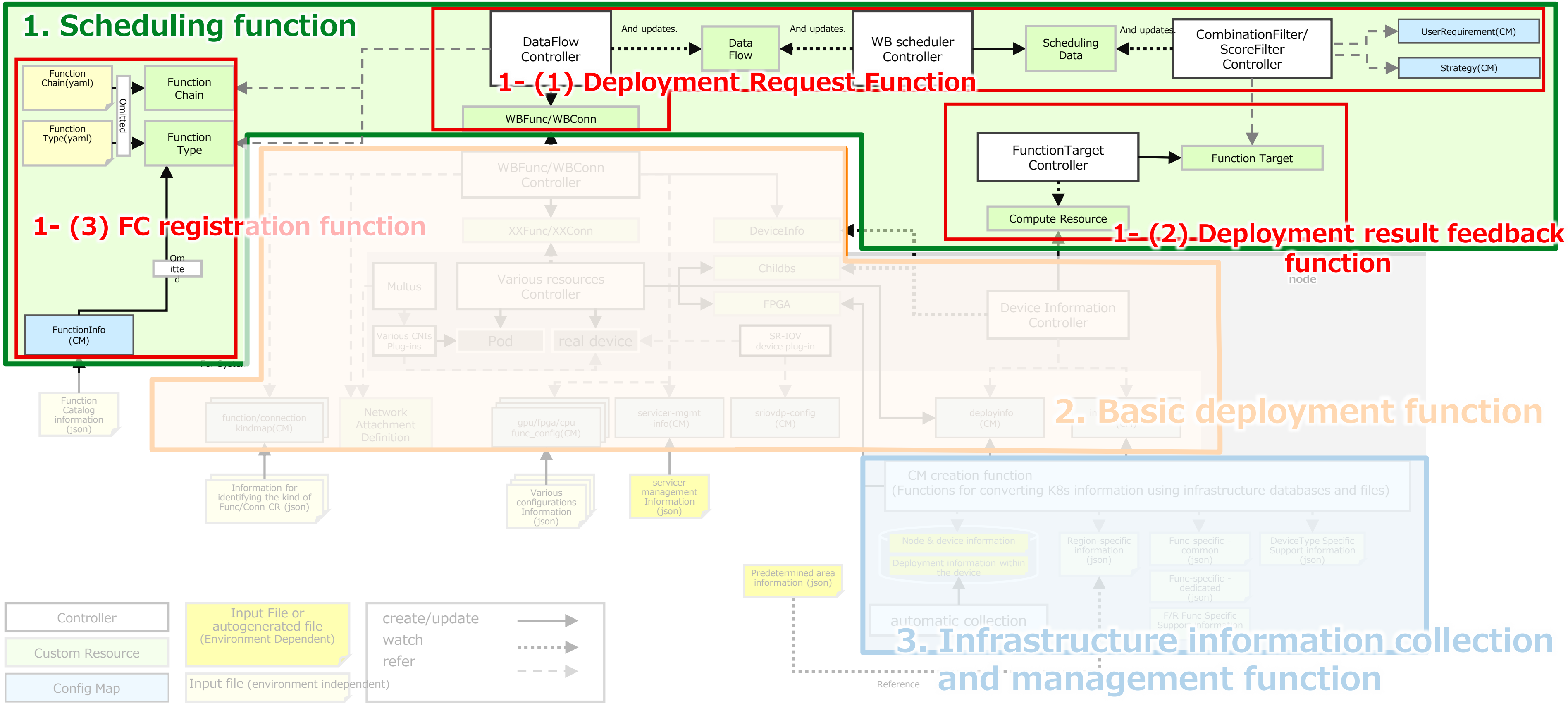
Three functions of the OpenKasugai controller (Reprinted)

Function name	Description
Scheduling function	• Automatically determines where each data flow component will be deployed in response to data flow deployment requests, taking into account scheduling condition selected by the user and current usage • Demand that basic deployment functions deploy each component of data flow (WBFunction/WBConnection) • You can register FC templates and associated functions and connections.
Basic deployment function	• In response to deployment requests for each data flow component, use various resource controllers to actually deploy and configure the components. • Dynamically writes child bs if child bs is not written to the destination FPGA • The controller automatically completes the detailed parameters required for the various FPGA circuit and Pod. For example, channel number, IP address, port number • The results of deployment are provided to scheduling function as the usage status of various resources.
Infrastructure information collection and management function	• Provide automatic collection of infrastructure information and automatic creation of K8s resources 　• Automate the collection and use of environment-dependent information such as infrastructure configuration information and written area information on k8s 　• Auto-complete DF dependent information (different information for each DF)

Scheduling function

- Automatically determines where each data flow component will be deployed in response to data flow deployment requests, taking into account the user's choice of scheduling condition and current usage
    - Overview of the Scheduling Process
        1. Create a combination pattern of deployment destinations for each of the functions that make up DF from the function deployment destination candidates.
            - If there is an FPGA in which the circuit has been written and the number of DF accommodated is sufficient, select the deployment destination assuming that the circuit will be used.
            - If there is an unwritten FPGA and child bs can be written, select the deployment destination assuming a new write (select the circuit after writing as the deployment destination).
            - If the existing Pod is running and there is room for more DF, select the deployment location based on the assumption that the pod will be used.
            - If CPU/GPU is available and Pod can be created, select a deployment location assuming new pod creation
        2. Filter and score matching destination combination patterns, taking into account scheduling condition selected by the user and current usage
            - Retry if there are no candidates in the middle
        3. Select one of the combination patterns for deployment based on the score, and decide where to deploy each of the functions that make up DF.
- Demand that basic deployment functions deploy each component of data flow (WBFunction/WBConnection)
- You can register FC templates and associated functions and connections.

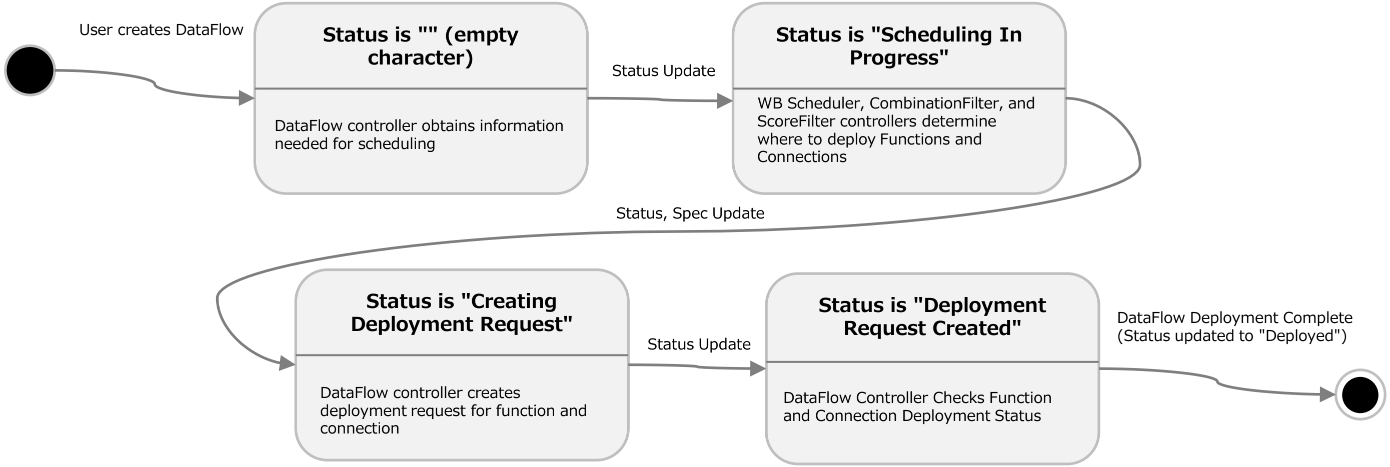
Scheduling function:
Functional configuration


1. Scheduling function
1- (1) Deployment Request Function
DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
WBFunc/WBConn
1- (3) FC registration function
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
Om itte d
FunctionInfo (CM)
Function Catalog information (json)
1- (2) Deployment result feedback function
FunctionTarget Controller
Function Target
Compute Resource
node
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Device Information Controller
function/connection kindmap(CM)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
2. Basic deployment function
Information for identifying the kind of Func/Conn CR (json)
Various configurations Information (json)
servicer management Information (json)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information
DeviceType Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
3. Infrastructure information collection and management function
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

Scheduling function:
Deployment request function (Overview)

- With the creation of DataFlow, custom resource, it schedules the deployment destination and sends a deployment request to basic deployment functions.
- Depending on DataFlow status (*1), DataFlow controller, WB scheduler controller,CombinationFilter and ScoreFilter controllers work together

(*1) Character string of DataFlow's Status.Status field



User creates DataFlow

Status is "" (empty character)

DataFlow controller obtains information needed for scheduling

Status Update

Status is "Scheduling In Progress"

WB Scheduler, CombinationFilter, and ScoreFilter controllers determine where to deploy Functions and Connections

Status, Spec Update

Status is "Creating Deployment Request"

DataFlow controller creates deployment request for function and connection

Status Update

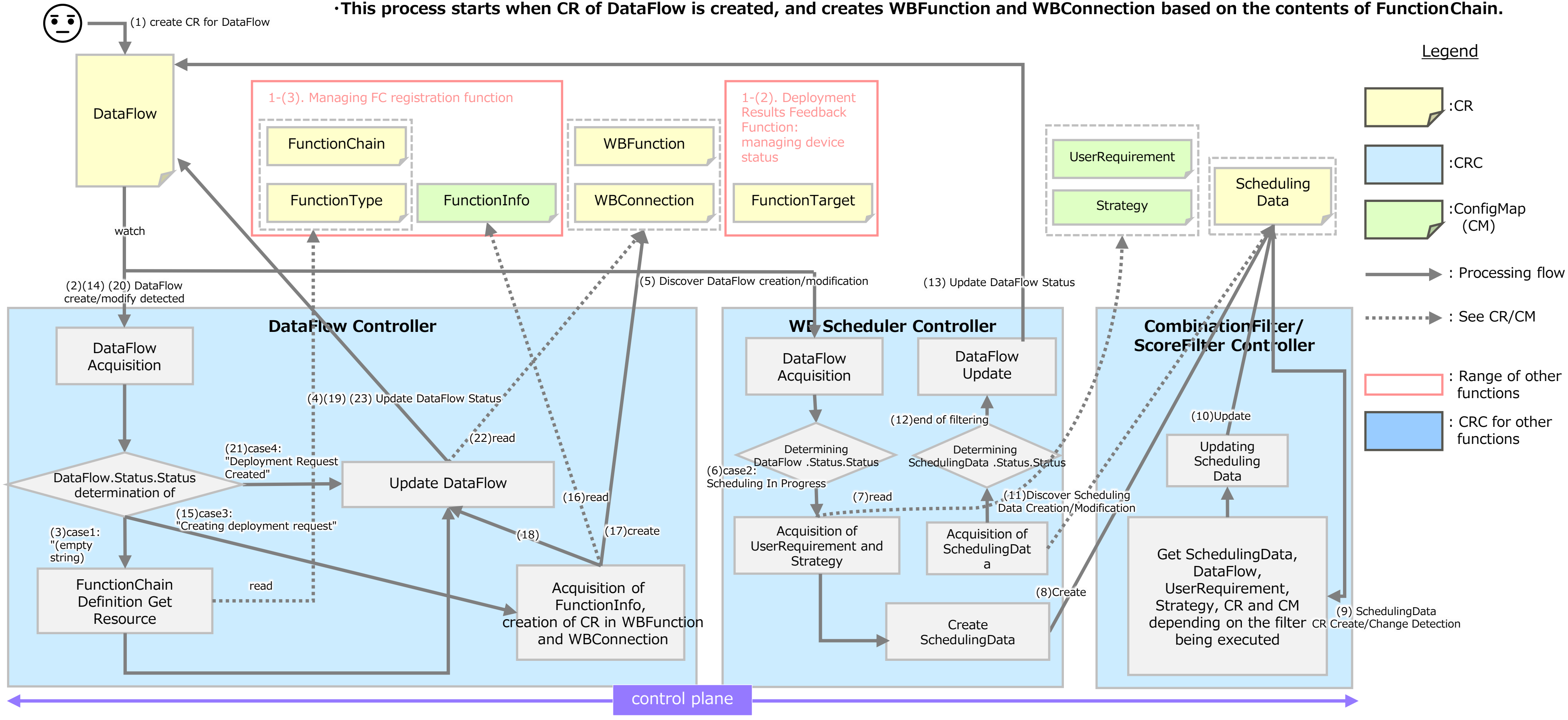
Status is "Deployment Request Created"

DataFlow Controller Checks Function and Connection Deployment Status

DataFlow Deployment Complete (Status updated to "Deployed")

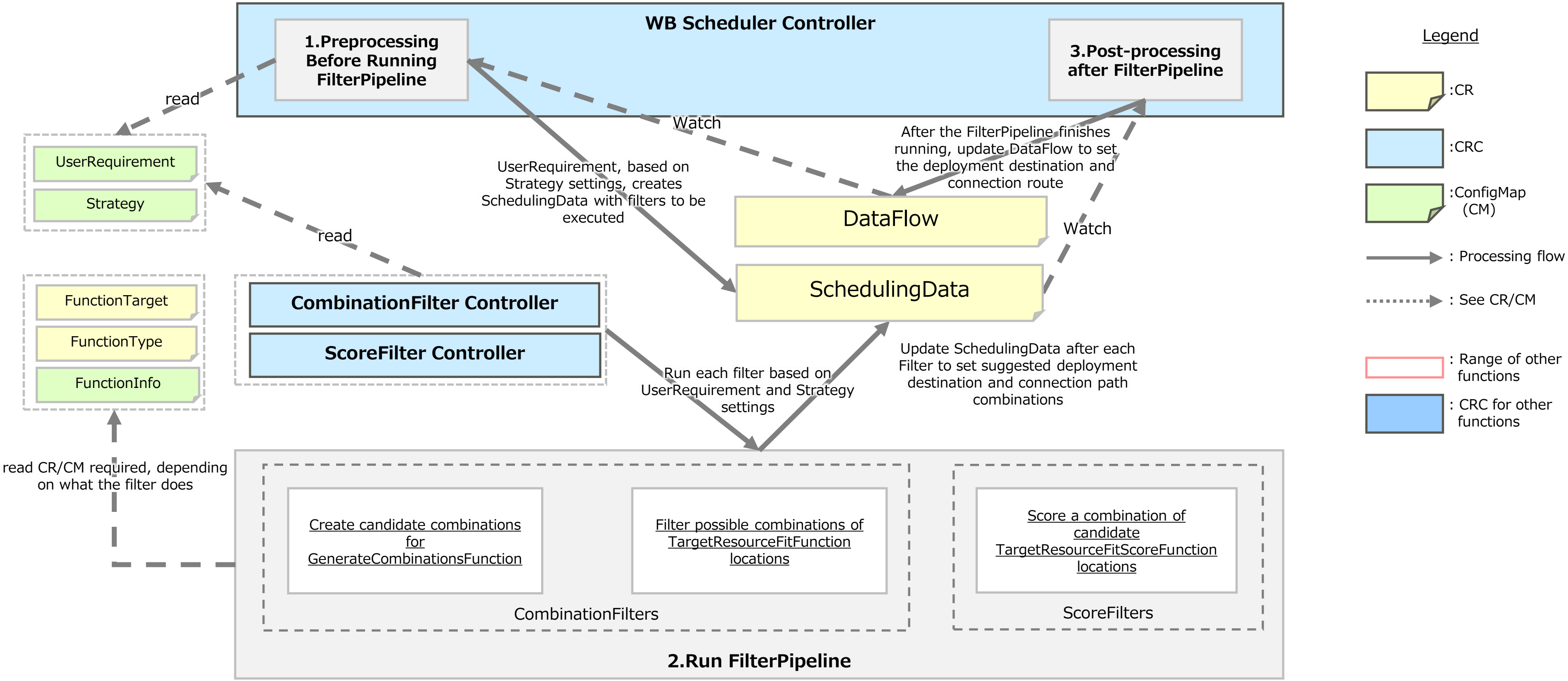
Scheduling function:
Deployment request function (Block diagram)

・This process starts when CR of DataFlow is created, and creates WBFunction and WBConnection based on the contents of FunctionChain.



(1) create CR for DataFlow
DataFlow
watch
(2)(14) (20) DataFlow create/modify detected
1-(3). Managing FC registration function
FunctionChain
FunctionType
FunctionInfo
WBFunction
WBConnection
1-(2). Deployment Results Feedback Function: managing device status
FunctionTarget
UserRequirement
Strategy
Scheduling Data
(5) Discover DataFlow creation/modification
(13) Update DataFlow Status
DataFlow Controller
DataFlow Acquisition
(4)(19) (23) Update DataFlow Status
(22)read
(21)case4: "Deployment Request Created"
DataFlow.Status.Status determination of
Update DataFlow
(16)read
(15)case3: "Creating deployment request"
(3)case1: "(empty string)
(18)
(17)create
FunctionChain Definition Get Resource
read
Acquisition of FunctionInfo, creation of CR in WBFunction and WBConnection
WB Scheduler Controller
DataFlow Acquisition
DataFlow Update
(12)end of filtering
Determining DataFlow .Status.Status
Determining SchedulingData .Status.Status
(6)case2: Scheduling In Progress
(7)read
(11)Discover Scheduling Data Creation/Modification
Acquisition of UserRequirement and Strategy
Acquisition of SchedulingDat a
(8)Create
Create SchedulingData
CombinationFilter/ ScoreFilter Controller
(10)Update
Updating Scheduling Data
Get SchedulingData, DataFlow, UserRequirement, Strategy, CR and CM depending on the filter being executed
(9) SchedulingData CR Create/Change Detection
control plane
Legend
:CR
:CRC
:ConfigMap (CM)
: Processing flow
: See CR/CM
: Range of other functions
: CRC for other functions

Scheduling function: Deployment request function (Scheduling block diagram)



WB Scheduler Controller

1.Preprocessing Before Running FilterPipeline

3.Post-processing after FilterPipeline

read

UserRequirement

Strategy

Watch

UserRequirement, based on Strategy settings, creates SchedulingData with filters to be executed

After the FilterPipeline finishes running, update DataFlow to set the deployment destination and connection route

DataFlow

Watch

read

SchedulingData

FunctionTarget

FunctionType

FunctionInfo

CombinationFilter Controller

ScoreFilter Controller

Update SchedulingData after each Filter to set suggested deployment destination and connection path combinations

Run each filter based on UserRequirement and Strategy settings

read CR/CM required, depending on what the filter does

Create candidate combinations for GenerateCombinationsFunction

Filter possible combinations of TargetResourceFitFunction locations

Score a combination of candidate TargetResourceFitScoreFunction locations

CombinationFilters

ScoreFilters

2.Run FilterPipeline

Legend

- :CR
- :CRC
- :ConfigMap (CM)
- : Processing flow
- : See CR/CM
- : Range of other functions
- : CRC for other functions

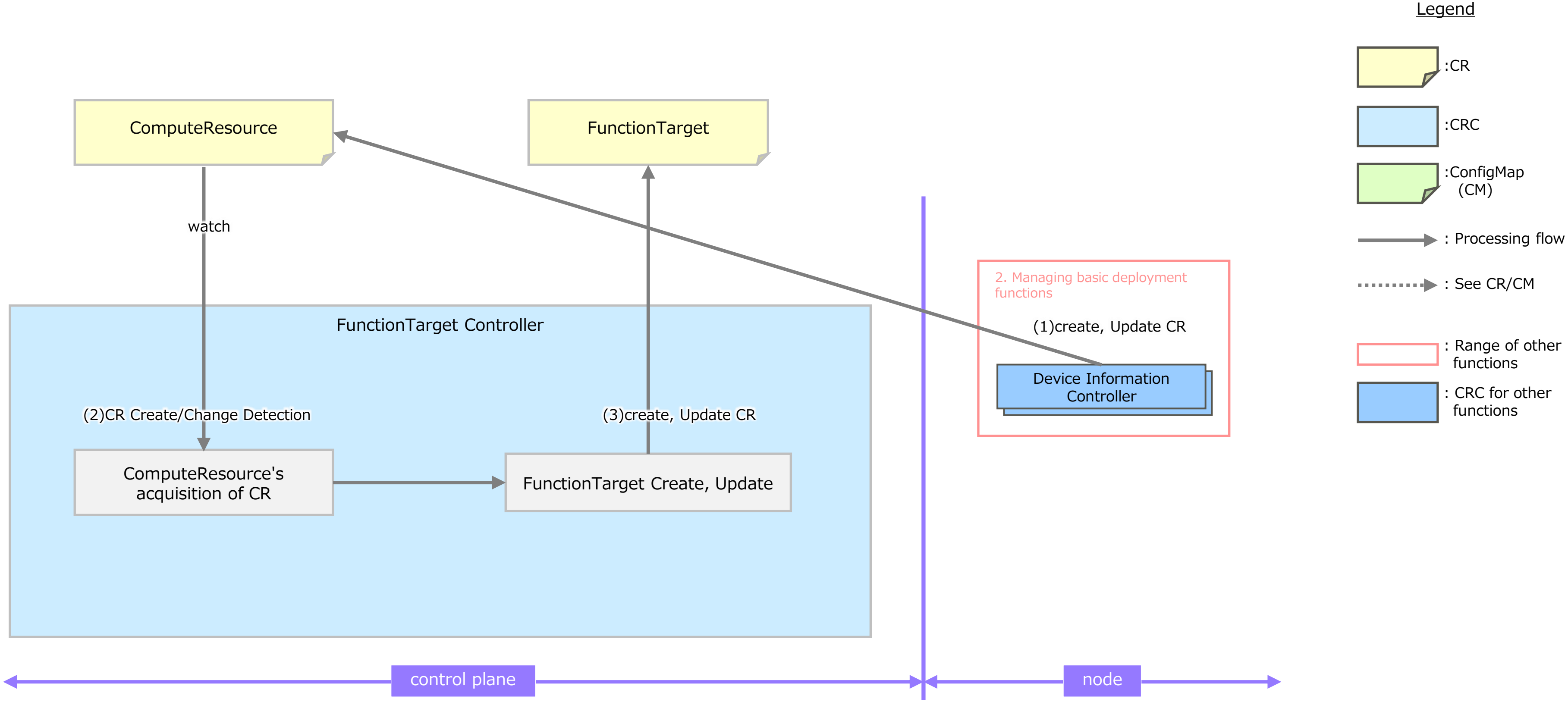
Scheduling function:
Deployment results feedback function (Overview)

- Feedback to scheduler on device usage as a result of basic deployment functions deploying WBFunction
  - Ability to create, update, and delete device information in accordance with the creation, update, and deletion of ConfigMap (ComputeResource) for creating device information
  - Use a FunctionTarget controller as a custom controller to manage the above device information.
    Manage device information as Status information of FunctionTarget, which is custom resource that is referenced during scheduling.

- Reference
  - FunctionTarget: custom resource to represent where a Function is deployed
    - Manages the area (= Lane. Device itself if there are no multiple areas) on the device where the Function can be deployed
    - If more than one area can be deployed, such as multiple channels or multiple Pod, this information is managed as a list in FunctionTarget .functions parameter
    - See CR/CM Specifications for details

Scheduling function:
Deployment results feedback function (Block Diagram)

・This process operates when CR is created or updated, and performs FunctionTarget creation or update based on the contents of ComputeResource.

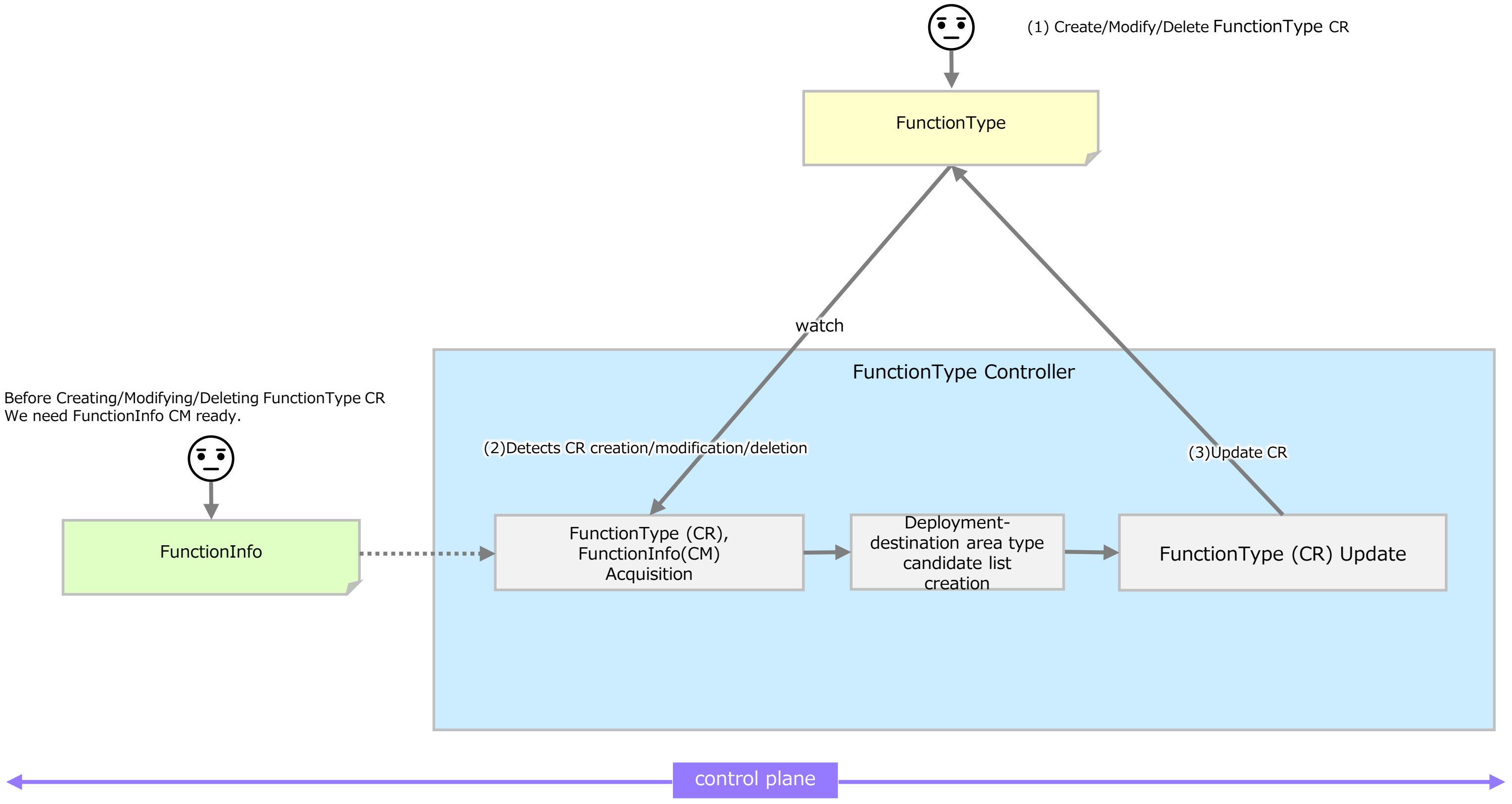

Legend
:CR
:CRC
:ConfigMap (CM)
: Processing flow
: See CR/CM
: Range of other functions
: CRC for other functions
ComputeResource
FunctionTarget
watch
FunctionTarget Controller
(2)CR Create/Change Detection
(3)create, Update CR
ComputeResource's acquisition of CR
FunctionTarget Create, Update
2. Managing basic deployment functions
(1)create, Update CR
Device Information Controller
control plane
node

Scheduling function:
FC registration function

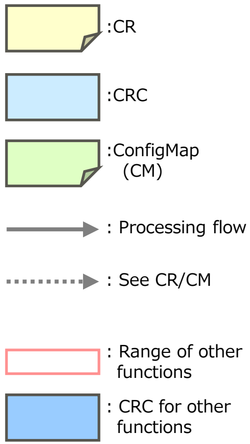
- Perform the processing necessary for registering and managing FunctionChain templates
  - Registering and changing the function catalog (FunctionType CR)
    - Fetch FunctionInfo and its own data, check the deployableItems value, and edit FunctionType's deployment suggestions
  - Registration and change of function chain definition (FunctionChain CR)
    - Get FunctionInfo/FunctionType and their own data and check for registration of functions defined in FunctionChain (FunctionType is Ready).
      Also check if there is an error over the number of connections of Function or duplicate connection Port.

Scheduling function:
FC registration function (Block diagram)

・This process works when CR of FunctionType is created, updated, or deleted, and updates CR of FunctionType based on the contents of Configmap of FunctionInfo.


(1) Create/Modify/Delete FunctionType CR
FunctionType
watch
FunctionType Controller
Before Creating/Modifying/Deleting FunctionType CR
We need FunctionInfo CM ready.
(2)Detects CR creation/modification/deletion
(3)Update CR
FunctionInfo
FunctionType (CR), FunctionInfo(CM) Acquisition
Deployment-destination area type candidate list creation
FunctionType (CR) Update
control plane

Legend



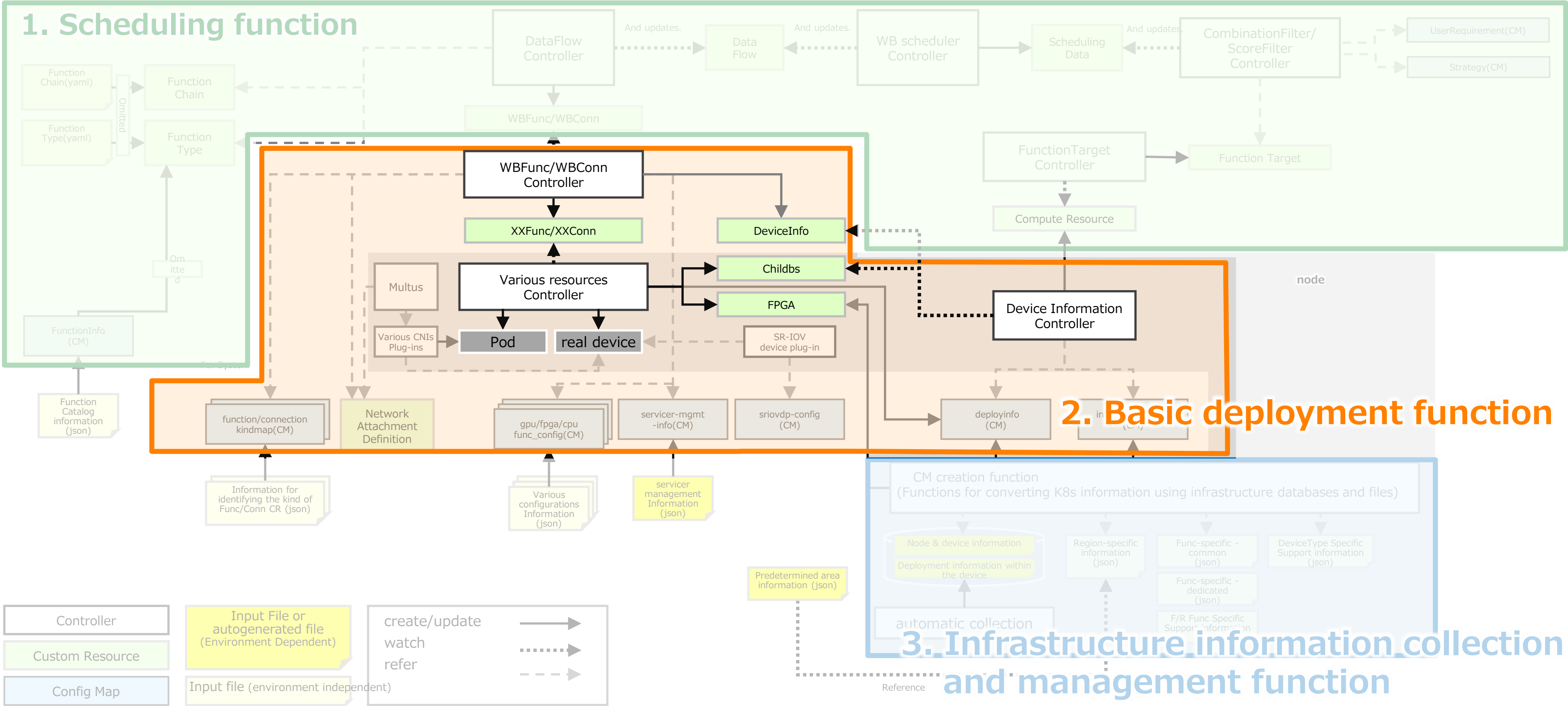
Basic deployment functions

- In response to deployment requests for data flow components, the various resource controllers are used to actually deploy and configure the components.
- Dynamically writes child bs if child bs is not written to the destination FPGA
- Detailed parameters required for various FPGA circuit and Pod are automatically complemented by the controller
  - For example, channel number, IP address, port number
- The results of deployment are provided to scheduling function as the usage status of various resources.

Basic deployment function:
Basic policy for information management of infrastructure and apps

- Define common items in the configuration information and use that information when creating various functions.
  - "Common items" are items that have the same value in any DF as long as they are the function and CR Kind of the configuration information.
    - Example:Function files (container image names, bs file names, etc.) and routine parts under pod "Template" in CPUFunction/GPUFunction, etc.
- Necessary name information for individual items is managed by the system side and automatically complemented by basic deployment functions section.
  - Individual items are items whose values differ for each DF. or infrastructure configuration dependent items
    - Example:Items that have different values depending on the deployment destination environment (such as differences in infrastructure configuration and shared methods), items that do not depend on the environment but have different values for each DF, etc.
  - Automatic completion based on DF information provided by scheduling function Department and infrastructure information collected by the infrastructure information collection and management function

Basic deployment function:
Functional configuration

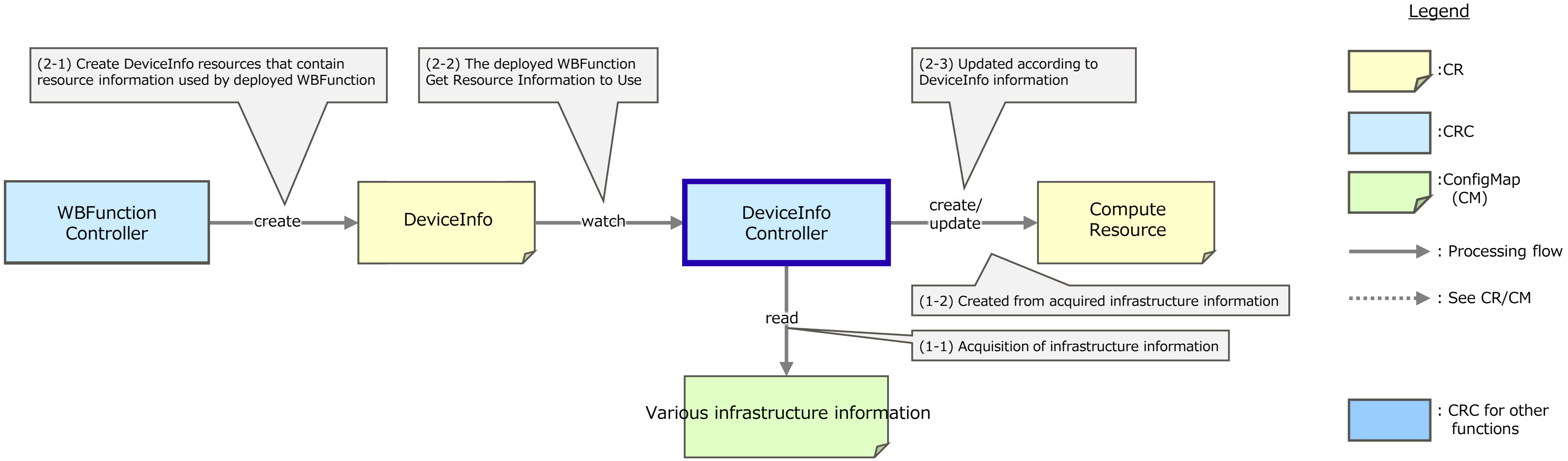

1. Scheduling function
DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Chain
Omitted
Function Type(yaml)
Function Type
WBFunc/WBConn
FunctionTarget Controller
Function Target
Compute Resource
Omitted
FunctionInfo (CM)
For System
Function Catalog information (json)
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Device Information Controller
node
function/connection kindmap(CM)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
in (CM)
2. Basic deployment function
Information for identifying the kind of Func/Conn CR (json)
Various configurations Information (json)
servicer management Information (json)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information
DeviceType Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
3. Infrastructure information collection and management function
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

Basic deployment function: Functional configuration (Described)

Item	Summary
Device Information Controller DeviceInfo controller	Create an Initial ComputeResource Resource (Device Information) Update ComputeResource based on deployment status
WBFunction Controller	Create CR for each function (CPUFunction/GPUFunction/FPGAFunction) from WBFunction resources Require the Device Information Controller to issue information about the devices required for deployment and update the deployment status as WBFunction is deployed
CPUFunction Controller	Deploy and manage Pod that runs the CPU
GPUFunction Controller	Deploy and manage GPU-powered Pod
FPGAFunction Controller	Deploy and manage the processing to be executed by the FPGA accelerator. Manages information and status of the FPGA and child bs, allocates resources within the FPGA and writes child bs to the FPGA as needed
WBConnection Controller	Creating CR for each connection system (EthernetConnection/PCIeConnection) from WBConnection resources
EthernetConnection Controller	Enable Ethernet connectivity between CPU, pod on GPU and external devices
PCIConnection Controller	Provide PCIe connectivity between accelerator (CPU, GPU, FPGA) devices

Basic deployment function: DeviceInfo controller

- DeviceInfo Controller (Device Information Controller) for infrastructure management is introduced to implement infrastructure resource management.
    - Obtain information on various devices and networks during system construction, and create ComputeResource for use in scheduling function Department
        - Based on various types of infrastructure information created by the Infrastructure Information Collection and Management Department, identify "areas" to be deployed, circuits and Pod already deployed in each area, and manage specifications and usage.
    - Update ComputeRsource with information from WBFunction controller when DF is deployed



(2-1) Create DeviceInfo resources that contain resource information used by deployed WBFunction

(2-2) The deployed WBFunction Get Resource Information to Use

(2-3) Updated according to DeviceInfo information

WBFunction Controller

create

DeviceInfo

watch

DeviceInfo Controller

create/ update

Compute Resource

(1-2) Created from acquired infrastructure information

read

(1-1) Acquisition of infrastructure information

Various infrastructure information

Legend

🟨 :CR

🟦 :CRC

🟩 :ConfigMap (CM)

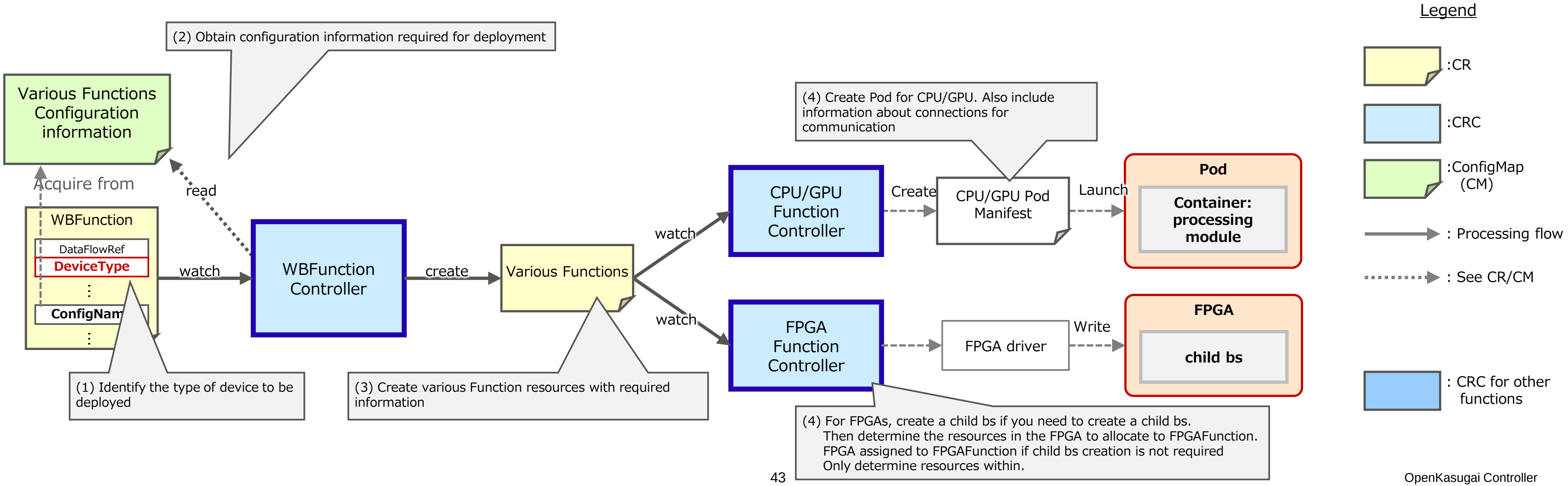
→ : Processing flow

⇢ : See CR/CM

🟦 : CRC for other functions

Basic deployment function: WBFunction controllers, various Function controllers

- Obtain information necessary to deploy WBFunction (Configuration information and device information) and create CR for each function type (CPUFunction/GPUFunction/FPGAFunction) corresponding to the device to be deployed.
- In response to CR of various Functions, the corresponding Function controller operates. Deploy various processing modules
  - Create Pod for CPU/GPU
  - In case of FPGA, child bs writing and circuit setting are performed via FPGA driver (details on next page)



(2) Obtain configuration information required for deployment

(4) Create Pod for CPU/GPU. Also include information about connections for communication

(1) Identify the type of device to be deployed

(3) Create various Function resources with required information

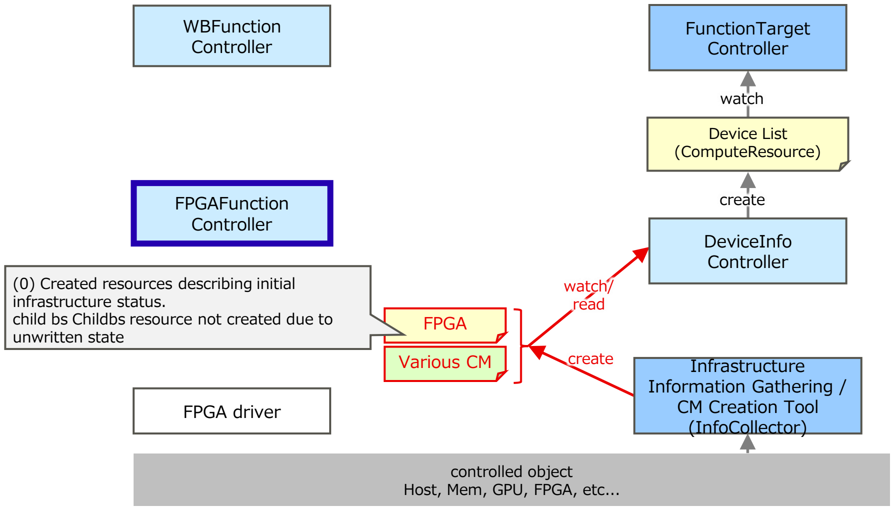
(4) For FPGAs, create a child bs if you need to create a child bs.
Then determine the resources in the FPGA to allocate to FPGAFunction.
FPGA assigned to FPGAFunction if child bs creation is not required
Only determine resources within.

Basic deployment function: FPGAFunction controller

- After receiving FPGAFunction's CR, child bs is written as necessary, and various processing modules are deployed.
  - If the deployment destination is a written area, various settings are performed via the FPGA driver according to the issued device information.
  - If the deployment destination is an unwritten area, write child bs via the FPGA driver according to the issued device information, and then configure various settings

- Manage, create, and update the specifications and current status (writing) of FPGAs (FPGA resources) and child bs (Childbs) associated with deployment.

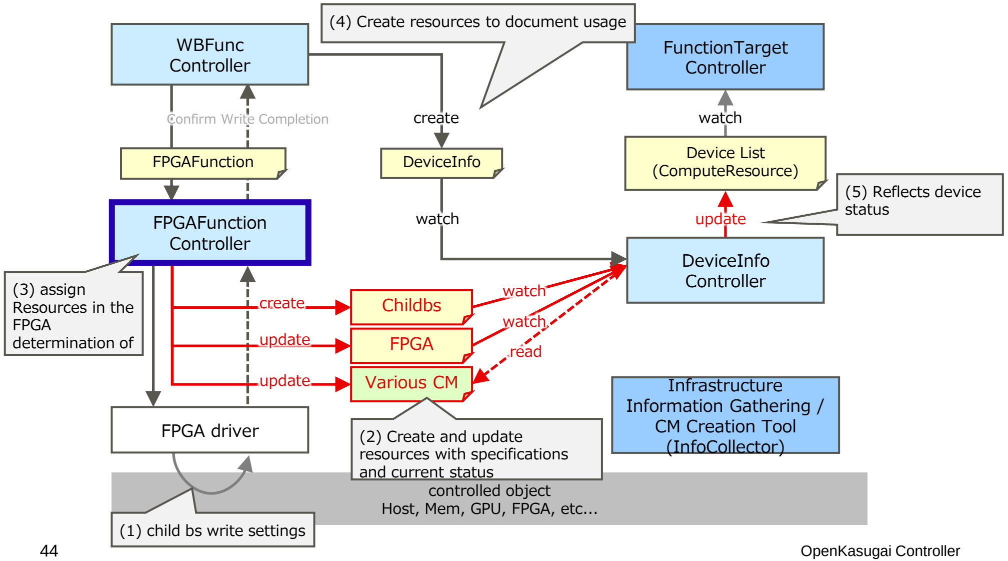
Reference: When building the system * Details will be described later.



WBFunction Controller
FunctionTarget Controller
watch
Device List (ComputeResource)
create
FPGAFunction Controller
DeviceInfo Controller
watch/ read
FPGA
Various CM
create
FPGA driver
Infrastructure Information Gathering / CM Creation Tool (InfoCollector)
controlled object
Host, Mem, GPU, FPGA, etc...

(0) Created resources describing initial infrastructure status.
child bs Childbs resource not created due to unwritten state

When deploying FPGAFunc with child bs writes



(4) Create resources to document usage

(5) Reflects device status

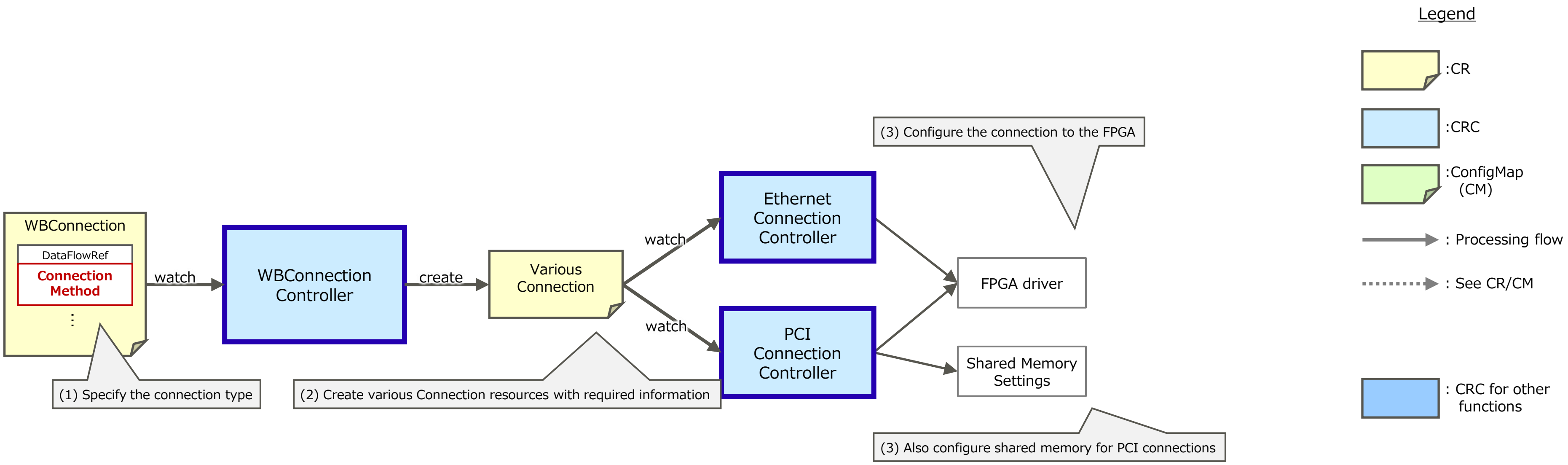
(3) assign Resources in the FPGA determination of

(2) Create and update resources with specifications and current status

(1) child bs write settings

Basic deployment function:
WBConnection controller, EthernetConnection/PCIConnection controller

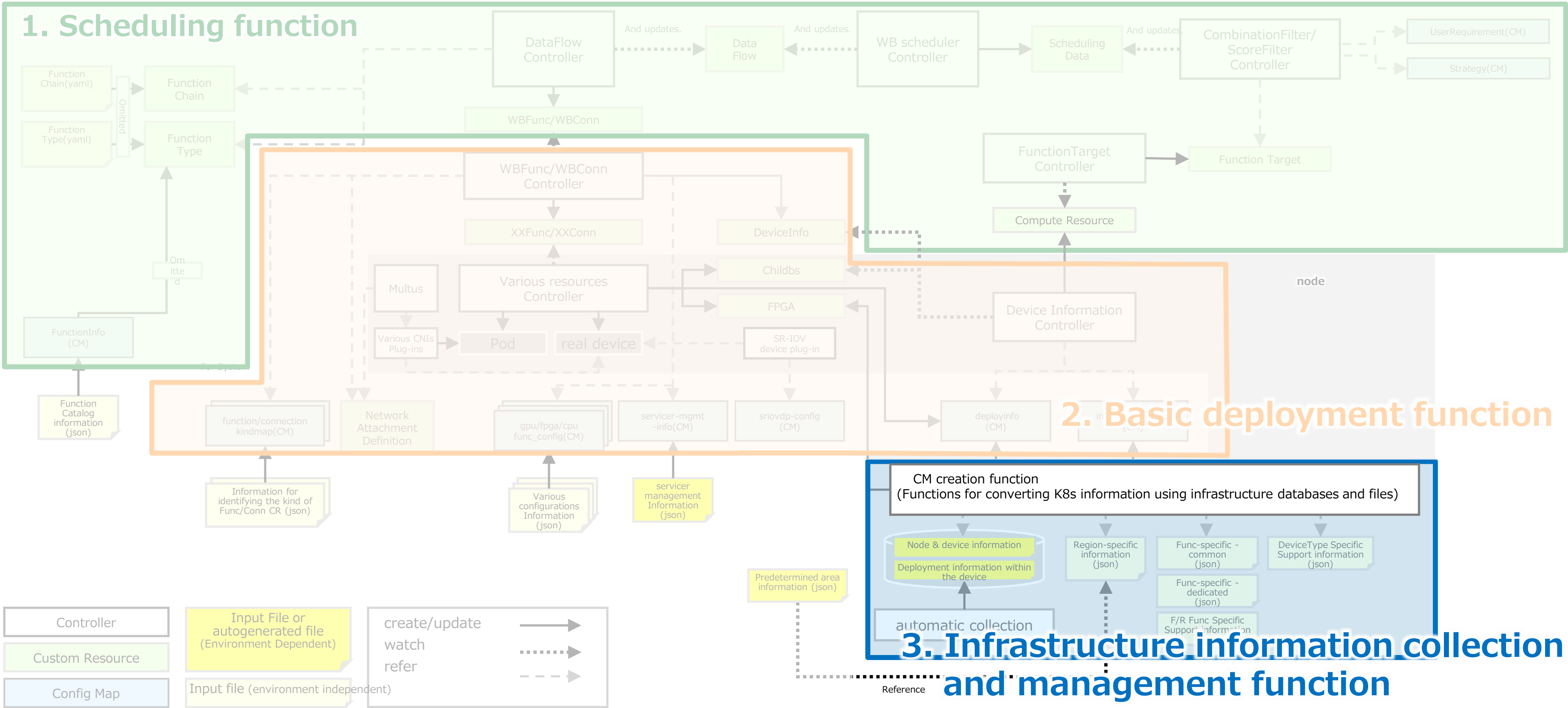
- Obtain information necessary for WBConnection deployment (Configuration information and device information), and create a CR for each connection type (EthernetConnection/PCIConnection).
- CR of each Connection causes the corresponding Connection controller to operate.
  - In the case of Ethernet, if the connection source/destination is an FPGA, set the connection via the FPGA driver.
  - For PCIe, configure shared memory for DMA, and if the source and destination are FPGAs, configure the same connection settings.
    - If the connection source/destination is CPU/GPU, this controller does nothing because the connection destination information is written in the manifest of Pod.


Legend
:CR
:CRC
:ConfigMap (CM)
: Processing flow
: See CR/CM
: CRC for other functions
WBConnection
DataFlowRef
Connection Method
⋮
watch
WBConnection Controller
create
Various Connection
watch
watch
Ethernet Connection Controller
PCI Connection Controller
FPGA driver
Shared Memory Settings
(1) Specify the connection type
(2) Create various Connection resources with required information
(3) Configure the connection to the FPGA
(3) Also configure shared memory for PCI connections

Infrastructure information collection and management function

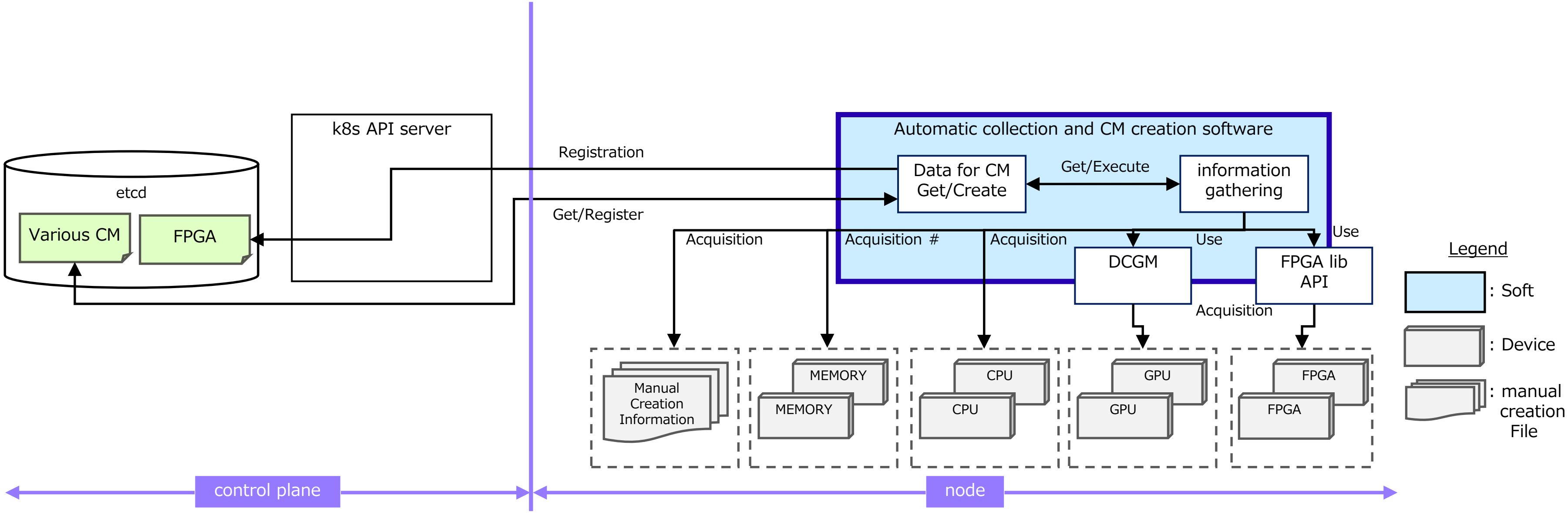
- Provide automatic collection of infrastructure information and automatic creation of K8s resources
  - Automate the collection and use of environment-dependent information such as infrastructure configuration information and written area information on k8s
    - Reduce the amount of effort required to build the environment
    - Facilitate lateral deployment to other environments
  - Auto-complete DF dependent information (different information for each DF)
    - Reduces user (DF deployment requestor) effort to prepare for DF deployment

Infrastructure information collection and management: Functional configuration


1. Scheduling function
DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
FunctionTarget Controller
Function Target
Compute Resource
Omitted
FunctionInfo (CM)
Function Catalog information (json)
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Device Information Controller
node
function/connection kindmap(CM)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
2. Basic deployment function
Information for identifying the kind of Func/Conn CR (json)
Various configurations Information (json)
servicer management Information (json)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information
DeviceType Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
3. Infrastructure information collection and management function
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

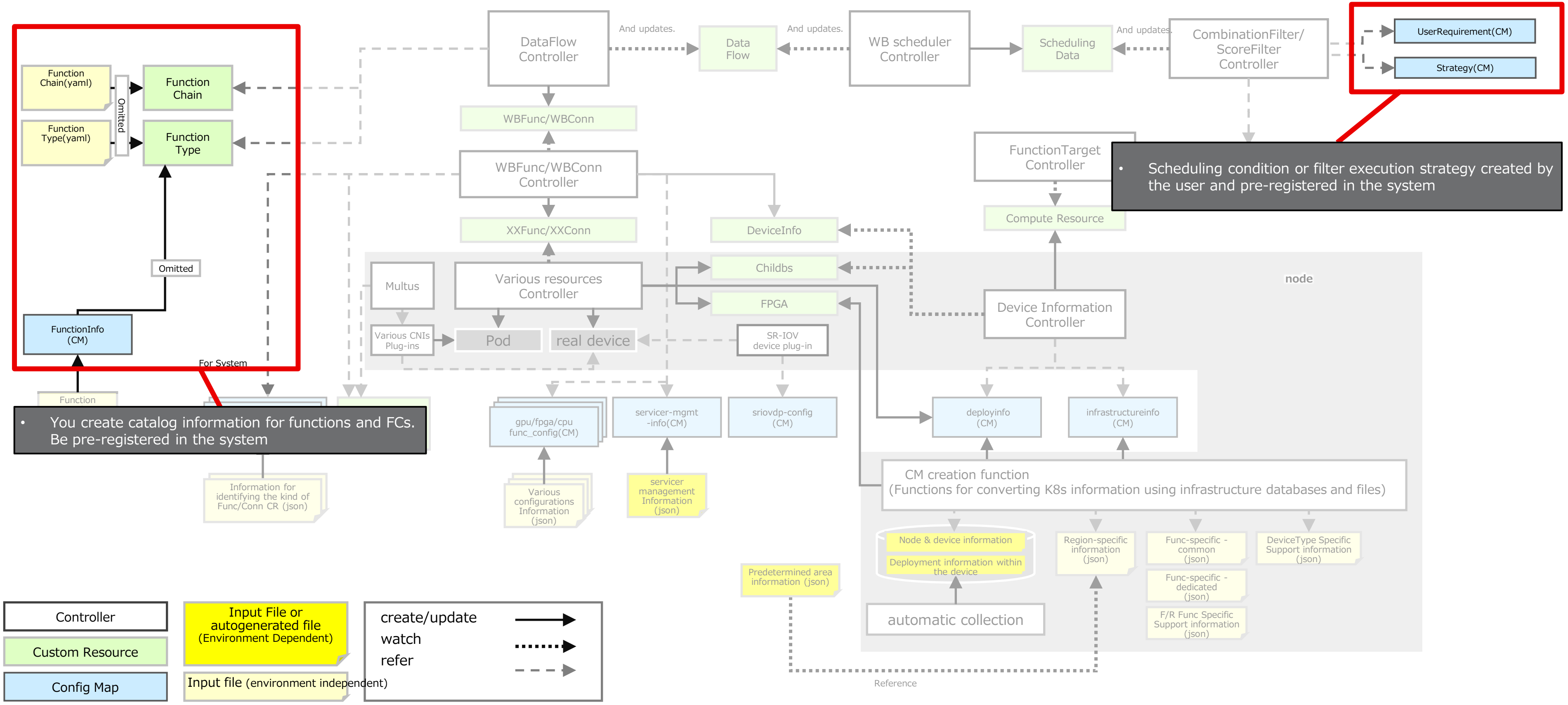
Infrastructure information collection and management: Automatic collection and CM creation software

- Collecting FPGA/GPU/CPU/memory device information
- Create and register ConfigMap/FPGA CR based on the collected device information



5. Overall flow of operations (With FPGA child bs write)

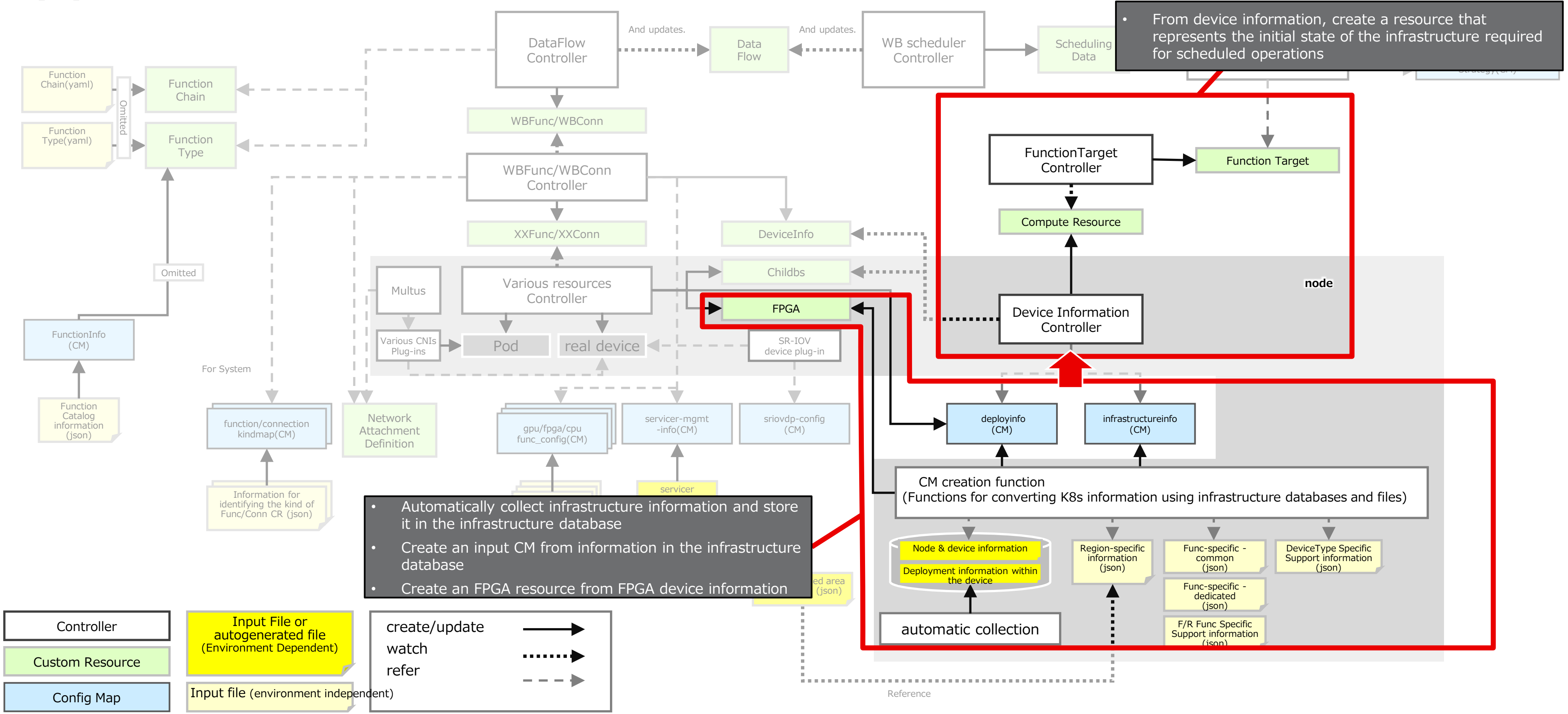
Overall flow of operations:
(1) Catalog information and scheduling condition registration


Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
Omitted
FunctionInfo (CM)
For System
Function
DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
FunctionTarget Controller
Compute Resource
Multus
Various resources Controller
Childbs
FPGA
node
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Device Information Controller
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
deployinfo (CM)
infrastructureinfo (CM)
Information for identifying the kind of Func/Conn CR (json)
Various configurations Information (json)
servicer management Information (json)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
DeviceType Specific Support information (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

- Scheduling condition or filter execution strategy created by the user and pre-registered in the system

- You create catalog information for functions and FCs. Be pre-registered in the system

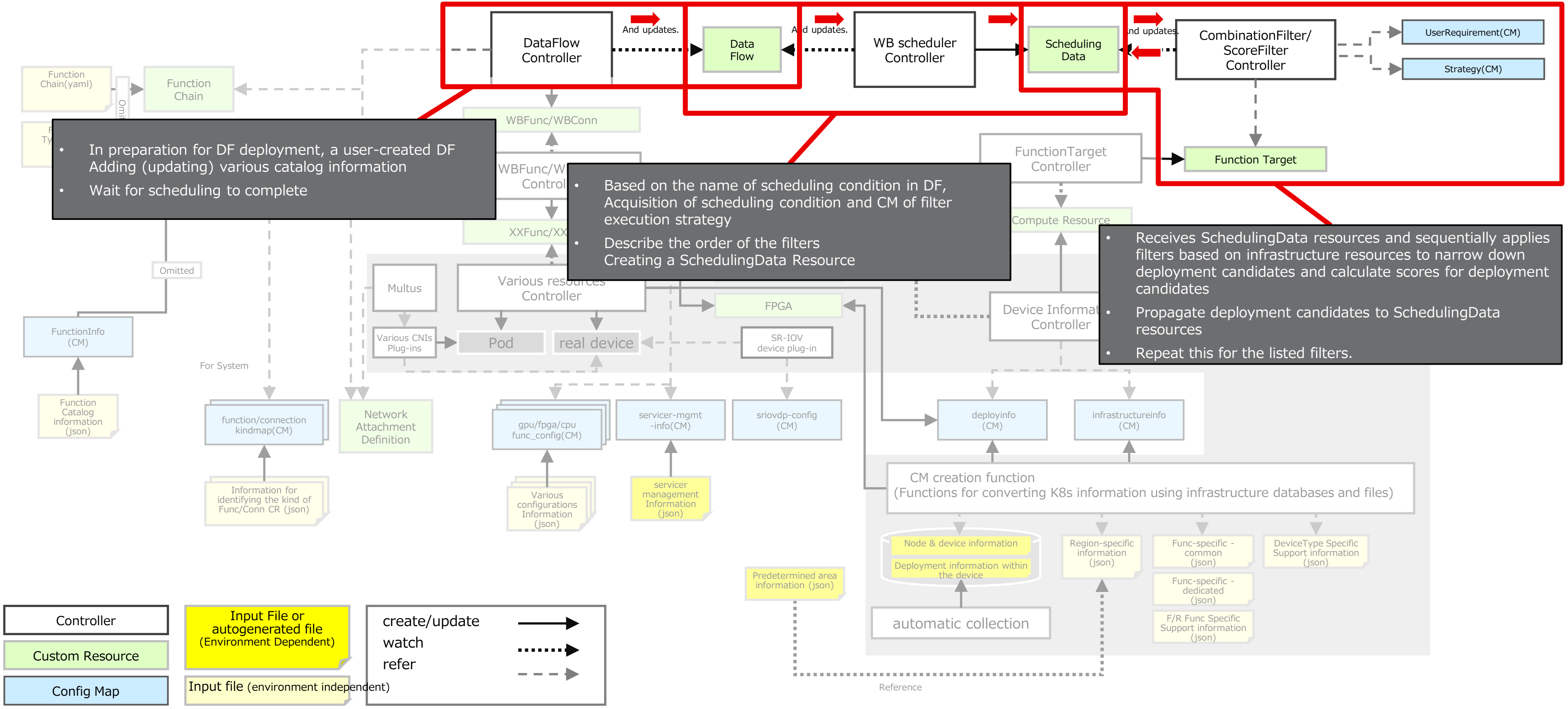
Overall flow of operations:
(2) Automatic collection of infrastructure information,
(3) Creation of initial infrastructure resources


DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
Omitted
FunctionInfo (CM)
Function Catalog information (json)
For System
function/connection kindmap(CM)
Information for identifying the kind of Func/Conn CR (json)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
servicer-mgmt -info(CM)
sriovdp-config (CM)
servicer
ed area (json)
FunctionTarget Controller
Function Target
Compute Resource
node
Device Information Controller
deployinfo (CM)
infrastructureinfo (CM)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
DeviceType Specific Support information (json)
automatic collection
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

- From device information, create a resource that represents the initial state of the infrastructure required for scheduled operations

- Automatically collect infrastructure information and store it in the infrastructure database
- Create an input CM from information in the infrastructure database
- Create an FPGA resource from FPGA device information

Overall flow of operations:
(4) Preparation for DF deployment, (5) Preparation for scheduling, (6) Scheduling



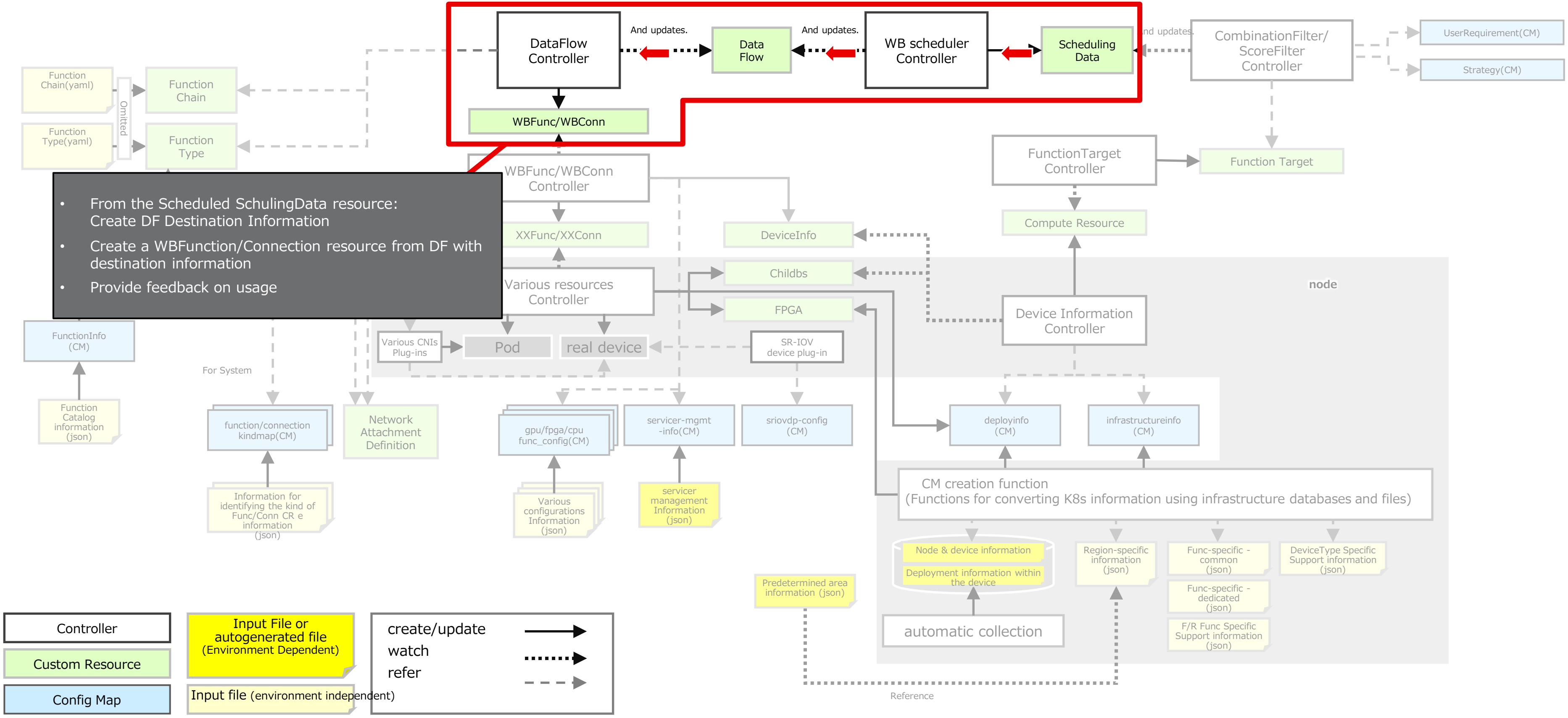
- In preparation for DF deployment, a user-created DF Adding (updating) various catalog information
- Wait for scheduling to complete

- Based on the name of scheduling condition in DF, Acquisition of scheduling condition and CM of filter execution strategy
- Describe the order of the filters Creating a SchedulingData Resource

- Receives SchedulingData resources and sequentially applies filters based on infrastructure resources to narrow down deployment candidates and calculate scores for deployment candidates
- Propagate deployment candidates to SchedulingData resources
- Repeat this for the listed filters.

DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Target
FunctionTarget Controller
Compute Resource
Device Information Controller
Function Chain(yaml)
Function Chain
Omitted
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
Omitted
FunctionInfo (CM)
Function Catalog information (json)
For System
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
FPGA
SR-IOV device plug-in
function/connection kindmap(CM)
Information for identifying the kind of Func/Conn CR (json)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
Various configurations Information (json)
servicer-mgmt -info(CM)
servicer management Information (json)
sriovdp-config (CM)
deployinfo (CM)
infrastructureinfo (CM)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
automatic collection
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
DeviceType Specific Support information (json)
Predetermined area information (json)
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

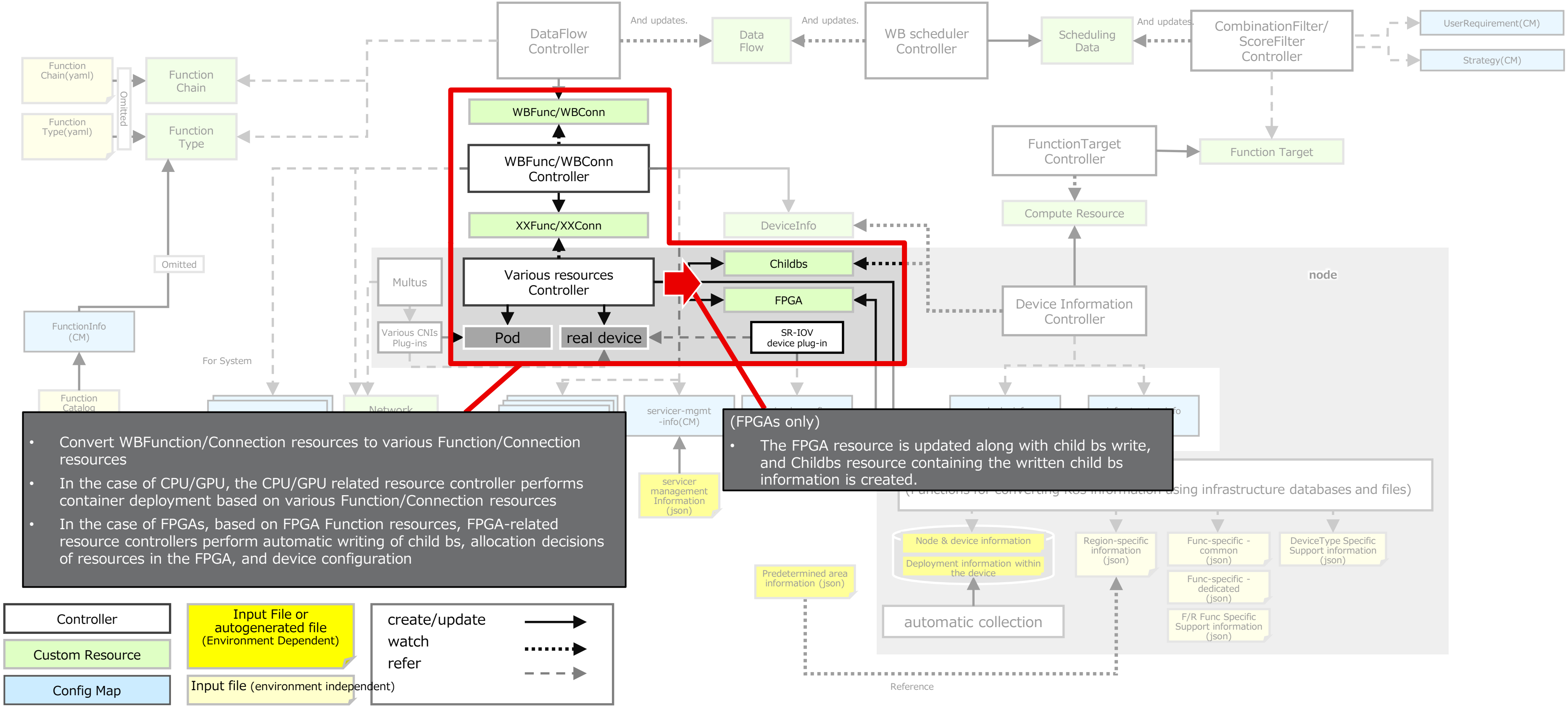
Overall flow of operations:
(7) Creating a WBFunction/Connection resource



DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
WBFunc/WBConn
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn Controller
XXFunc/XXConn
DeviceInfo
Childbs
FPGA
Various resources Controller
FunctionTarget Controller
Function Target
Compute Resource
node
Device Information Controller
FunctionInfo (CM)
Function Catalog information (json)
For System
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
function/connection kindmap(CM)
Information for identifying the kind of Func/Conn CR e information (json)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
Various configurations Information (json)
servicer-mgmt -info(CM)
servicer management Information (json)
sriovdp-config (CM)
deployinfo (CM)
infrastructureinfo (CM)
CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
DeviceType Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

- From the Scheduled SchulingData resource: Create DF Destination Information
- Create a WBFunction/Connection resource from DF with destination information
- Provide feedback on usage

Overall flow of operations:
(8) Deployment and setting of various functions

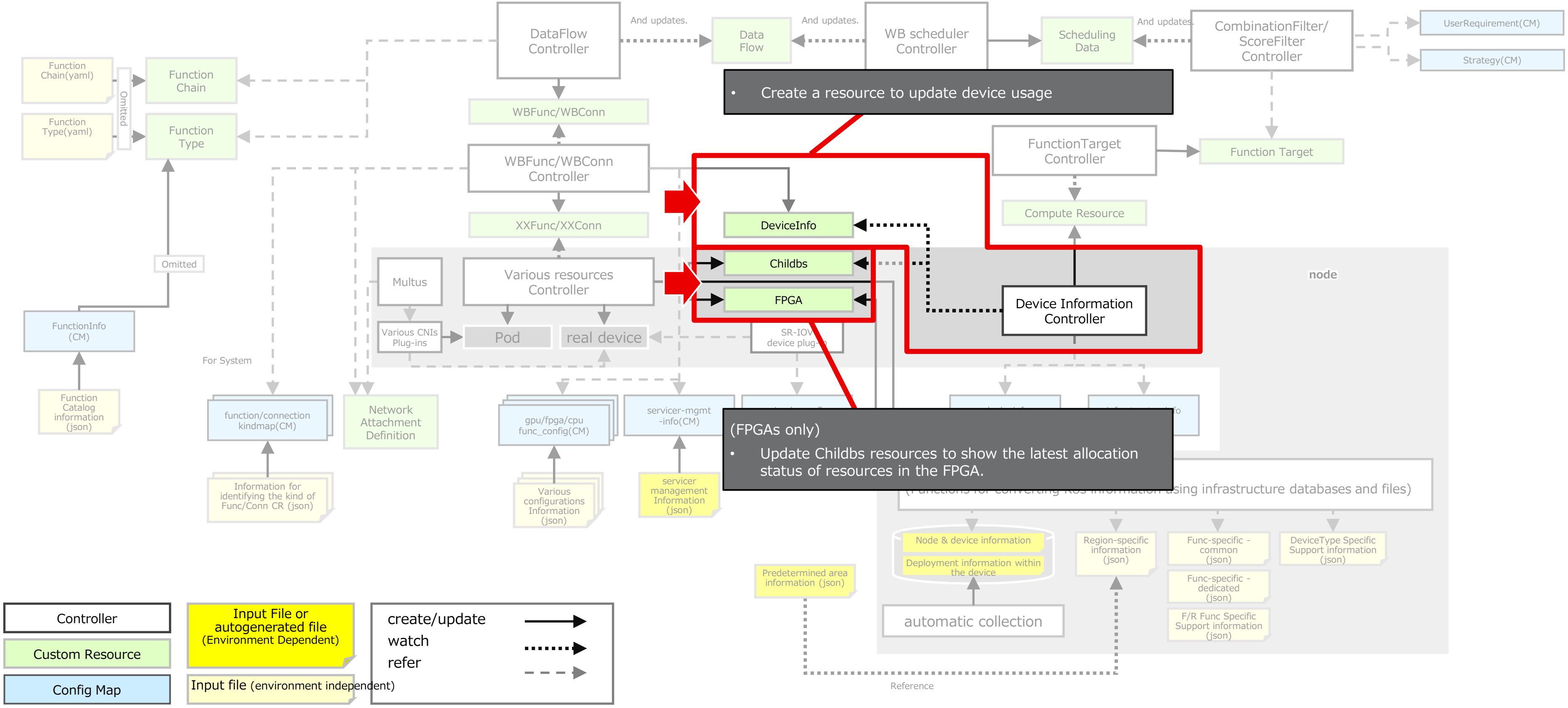

DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
FunctionTarget Controller
Function Target
Compute Resource
DeviceInfo
Childbs
FPGA
SR-IOV device plug-in
node
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
Device Information Controller
Omitted
FunctionInfo (CM)
For System
Function Catalog
Network
servicer-mgmt -info(CM)
servicer management Information (json)
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
DeviceType Specific Support information (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

- Convert WBFunction/Connection resources to various Function/Connection resources
- In the case of CPU/GPU, the CPU/GPU related resource controller performs container deployment based on various Function/Connection resources
- In the case of FPGAs, based on FPGA Function resources, FPGA-related resource controllers perform automatic writing of child bs, allocation decisions of resources in the FPGA, and device configuration

(FPGAs only)

- The FPGA resource is updated along with child bs write, and Childbs resource containing the written child bs information is created.

Overall flow of operations:
(9) Create and update resources for usage feedback



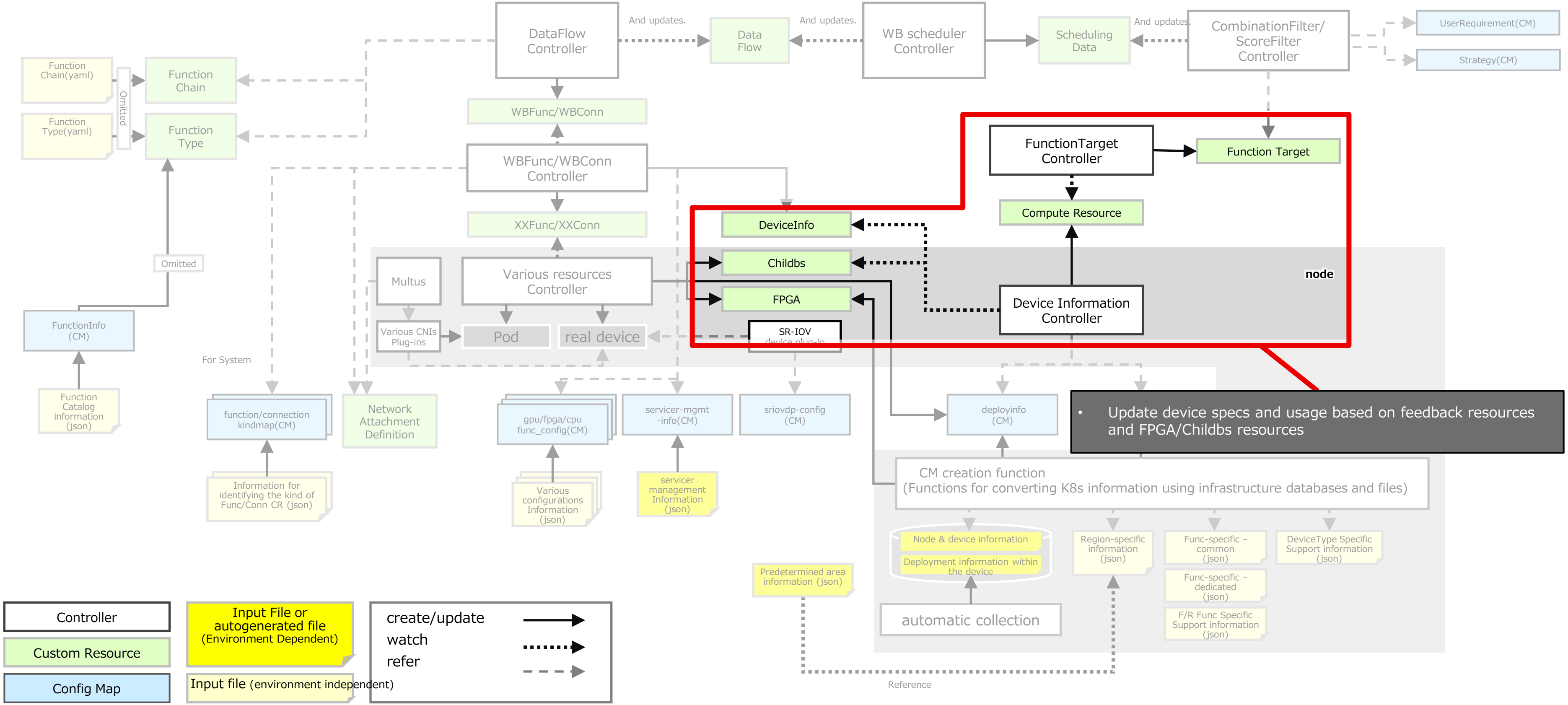
- Create a resource to update device usage

(FPGAs only)

- Update Childbs resources to show the latest allocation status of resources in the FPGA.

DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
FunctionTarget Controller
Function Target
Compute Resource
DeviceInfo
Childbs
FPGA
Device Information Controller
node
Omitted
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
SR-IOV device plug-in
FunctionInfo (CM)
Function Catalog information (json)
For System
function/connection kindmap(CM)
Information for identifying the kind of Func/Conn CR (json)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
Various configurations Information (json)
servicer-mgmt -info(CM)
servicer management Information (json)
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
Region-specific information (json)
Func-specific - common (json)
DeviceType Specific Support information (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
Predetermined area information (json)
automatic collection
Reference
Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer

Overall flow of operations:
(10) Feedback on usage of infrastructure resources



DataFlow Controller
And updates.
Data Flow
And updates.
WB scheduler Controller
Scheduling Data
And updates.
CombinationFilter/ ScoreFilter Controller
UserRequirement(CM)
Strategy(CM)
Function Chain(yaml)
Function Type(yaml)
Omitted
Function Chain
Function Type
WBFunc/WBConn
WBFunc/WBConn Controller
XXFunc/XXConn
Omitted
FunctionInfo (CM)
Function Catalog information (json)
For System
Multus
Various resources Controller
Various CNIs Plug-ins
Pod
real device
FunctionTarget Controller
Function Target
Compute Resource
DeviceInfo
Childbs
FPGA
SR-IOV device plug-in
Device Information Controller
node
function/connection kindmap(CM)
Information for identifying the kind of Func/Conn CR (json)
Network Attachment Definition
gpu/fpga/cpu func_config(CM)
Various configurations Information (json)
servicer-mgmt -info(CM)
servicer management Information (json)
sriovdp-config (CM)
deployinfo (CM)

- Update device specs and usage based on feedback resources and FPGA/Childbs resources

CM creation function
(Functions for converting K8s information using infrastructure databases and files)
Node & device information
Deployment information within the device
automatic collection
Region-specific information (json)
Func-specific - common (json)
Func-specific - dedicated (json)
F/R Func Specific Support information (json)
DeviceType Specific Support information (json)
Predetermined area information (json)
Reference

Controller
Custom Resource
Config Map
Input File or autogenerated file (Environment Dependent)
Input file (environment independent)
create/update
watch
refer