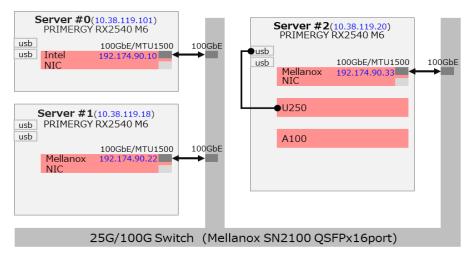
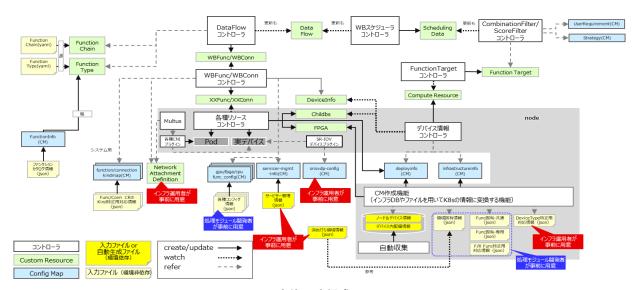
# OpenKasugai-Controller Install Manual(Attachment)

目次	説明
1.根定環境図など	OpenKasugai-Controller Install Manualにて想定している物理
1.忍足境境区及C	構成やソフト構成
2.YAML説明	設定内容についての解説
Z.YAML就明	補足事項シート有り
	設定内容や設定値についての解説。補足シートも参照のこと
3.CM作成に使用する入力データ(JSON)の説明	補足事項シート有り
4. CRCのdaemonset用YAML説明	CRCのdaemonset用YAMLの内容についての説明

v1.1.0



物理構成図



全体ソフト構成図

	ample-data/sample-data-demo/yaml/dataflows/test-ext-1/df-test-ext-1-1.yamlを例に説明する	
各処理モジュールのPodが使用するネットワーク情報(IPアドレス・ボート番号)の設 <b>DataFlow YAML</b> apiVersion: example.com/v1	た方法の注意については、「2.(補定).Datariówyamlの設定」を参照のこと 説明	備考
kind: DataFlow metadata:		
name: "df-test-3-1-1-1" namespace: "test01"	ユーザが任意に設定する ユーザが任意に設定する	
spec: functionChainRef: name: "cpu-decode-cpu-filter-resize-2types-high-infer-chain"	DataFlowが利用するFunctionChainのmetadata.Name	
namespace: "chain-imgproc" requirements:	DataFlowが利用するFunctionChainのmetadata.Namespace スケジューリング時に満たす必要がある要件を記載する	現在の試作ではファンクションチェイン全体の要件のみが指定可能(全てのファンクションに共通の要件が適用される)
all: capacity: 15 functionUserParameter:	ファンクジョンチェイン全体の要件を記載 現正負荷量(fps)を記載	各コネクションと各コネクションによる想定負荷量(fps)
- functionKey: decode-main userParams:	CPUデコードFunctionの識別子	
ipAddress: 192.174.90.101/24 inputPort: 5004 outputIPAddress: 192.174.90.111	自身のIPアドレス、Pod02nd NICOIPアドレスとして設定される 自身のボート番号 送儀先 (CPUフィルタリサイズ) のIPアドレス	SR-IOVのVFを作成した100GNICの物理IPアドレスと同一のサブネットのIPアドレスを指定する
outputPort: 15000 - functionKey: filter-resize-high-infer-main	ISSII	
userParams: ipAddress: 192.174.90.111/24	自身のIPアドレス、Podの2nd NICのIPアドレスとして設定される	SR-IOVのVFを作成した100GNICの物理IPアドレスと同一のサブネットのIPアドレスを指定する
inputPort: 15000 outputIPAddress: 192.174.90.121 outputPort: 16000	自身のボート番号 送偏先 (コピー分岐) のIPアドレス 送偏先 (コピー分岐) のボート番号	
- functionKey: copy-branch-main userParams:	コピー分岐Functionの識別子	
ipAddress: 192.174.90.121/24 inputIPAddress: 192.174.90.121 inputPort: 16000	自身のIPアドレス、Podの2nd NICのIPアドレスとして設定される 自身のIPアドレス、前回のファンクションとのTCP特徴の確立のために利用(サブネットマスクの設定は不要) 自身のボート番号	SR-10VのFを作成した100GNICの物理IPアドレスと同一のサブネットのIPアドレスを指定する 自身のipAddressの設立事態と同じIPアドレスを設立すれば似い(サブネットマスクの設立は不要)
branchOutputIPAddress: 192.174.90.141,192.174.90.142 branchOutputPort: 17000,18000	送機先(GPU高度推論1、GPU高度推論2)のIPアドレスをカンマ区切りで指定 送偏先(GPU高度推論1、GPU高度推論2)のボート番号をカンマ区切りで指定	
- functionKey: infer-1 userParams: ipAddress: 192.174.90.141/24	GPU高度推議Function1の協別子 自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される	SR-IOVのVFを作成した100GNICの物理IPPドレスと同一のサブネットのIPアドレスを指定する
inputIPAddress: 192.174.90.141 inputPort: 17000	自身のIPアドレス。GStreamerのビデオ処理のコマンド実行(fpqdepay)に利用(サブネットマスクの設定は不要) 自身のボート番号	3A-O-O-WYV-ST-MACL-100GHICU-M944-ア-ア-ウスに向 ベクジャやアーロスを加定する 自身のipAddressの設定値と同じIPアドレスを設定すれば良い(サブネットマスクの設定は不要)
outputIPAddress: 192.174.90.10 outputPort: 2001	送偏先(映像受傷ツール)のIPアドレス 送偏先(映像受傷ツール)のポート番号	
- functionKey: infer-2 userParams: ipAddress: 192.174.90.142/24	GPU調度推論Function2の識別子 自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される	SR-IDVのVFを作成した100GNICの物理IPアドレスと同一のサブネットのIPアドレスを搬定する
inputIPAddress: 192.174.90.142 inputPort: 18000	自身のIPアドレス。GStreamerのビデオ処理のコマンド実行(fpqdepay)に必要 自身のボート番号	自身のipAddressの設定値と同じIPアドレスを設定すれば強い
outputIPAddress: 192.174.90.10 outputPort: 2002	送偏先 (映像受傷ツール) のIPアドレス 送偏先 (映像受傷ツール) のボート番号	
userRequirement: user-requirement	DataFlowのスケシューリング用の各種設定情報の散得のために参照するUserRequirementのConfigMapの metadata.nameを指定	詳細は「OpenKasugai-Controller-InstallManual」の「9.7節 DataFlowのスケジューリング影響を設定する場合」を参照
FunctionChain YAML apiVersion: example.com/v1	說明	備考
kind: FunctionChain metadata:	7 4445661709973	
name: cpu-decode-cpu-filter-resize-2types-high-infer-chain namespace: chain-imgproc spec:	ユーザが任意に設定する ユーザが任意に設定する	
functionTypeNamespace: "wbfunc-imgproc" connectionTypeNamespace: "default"	FunctionType@Namespace ConnectionType@Namespace	現在の試作では未使用
functions:	FunctionChainを構成するFunctionのmap keyはConnectionsのFromまたはToて指定するFunctionの識別子(このFunctionChainリソースにおいてユニークな文字列。 各CROFunctionKeyに入る文字列)	
decode-main: functionName: "cpu-decode"	TEX-CON UNICOUNDED(L-XSS)、アカ) CPUデコードPUCKOの問題 F FunctionTypeSpecで定義したName, Versionを指定	
version: "1.0.0" filter-resize-high-infer-main:	CPUフィルタリサイズFunctionの識別子	
functionName: "cpu-filter-resize-high-infer" version: "1.0.0" copy-branch-main:	FunctionTypeSpecで定義したName, Versionを指定 CPUコピー分終Functionの識別子	
functionName: "copy-branch" version: "1.0.0"	FunctionTypeSpecで定義したName, Versionを指定	
infer-1: functionName: "high-infer"	GPU高度推論 (12目) Functionが識別子 - FunctionTypeSpecで定義したName, Versionを指定	
version: "1.0.0" infer-2: functionName: "high-infer"	GPU高度推論 (2つ目) Functionの識別子 FunctionTypeSpecで定義したName, Versionを指定	
version: "1.0.0" connections: - from:	FunctionChainを構成するConnectionのリスト	
- from:	Connectionにおける送儀元Functionの情報	・ConnectionのFromがFunctionChain(FC)の開始点(監視カメラなどのデータソース相当)の場合は、"wb-start-of-chain"が与始まる文字列を設定すること
functionKey: "wb-start-of-chain"  port: 0	データ送機元Functionの識別子、Functionsのmapのkey値を設定 データ送機元Functionの出力ボート識別番号(Functionが1出力の場合はDを指定)	・組合者が、シ州外人の打たて家の原に関始が実施を場合さ、やいきはからぐもかがり能と教情やグラ列を付与して、このでいりてユークな文字列を設定 予念と、例は、"からはからぐらわか」でからまむからぐわか。「ヤットのよれらぐらから、例と、"からまれらくらわかっな。"やかいまれらくたわかっかが、 ※別技、"からまれらくたわか"の後に付与さた表情や文字列は、解版に用いて、収力。
to: functionKey: "decode-main"	Connectionにおける送偏先Functionの情報 データ送偏先Functionの識別子。Functionsのmapのkey値を設定	
port: 0 connectionTypeName: "auto" - from:	デー分類機序Functionの入力ボート階別無荷Functionが1入力の場合はDを指定) ConnectionTypeのリンース名表とは"auto"を指定 Connectionに対ける影像元子unctionの情報	TCP/UDPのボード番号ではない 現在の試作は常に"auto"を指定。そのためConnectionTypeカスタムリソースを参照することはない
functionKey: "decode-main" port: 0	データ送偏元Functionの識別子。Functionsのmapのkey値を設定 データ送偏元Functionの出力ポート識別番号(Functionが1出力の場合はDを指定)	TCP/UDPのボート番号ではない
to: functionKey: "filter-resize-high-infer-main" port: 0	Connectionにおける送偏先Functionの情報 データ送偏先Functionの連邦子、Functionsのmapのkev線を設定	TCP/IDPのボート番号ではない
connectionTypeName: "auto" - from:	データ連備来Functionの入力ボート識別番号(Functionが1入力の場合は0を指定) ConnectionTypeのソンース名または"auto"を指定 ConnectionEX1お区場展でFunctionの情報	TCF/DOPOMIC-THEPS (Made) 現在の試作は常に"auto"を指定。そのためConnectionTypeカスタムリソースを参照することはない
functionKey: "filter-resize-high-infer-main" port: 0	データ送偏元Functionの識別子。Functionsのmapのkey値を設定 データ送偏元Functionの出力ボート進別番号(Functionが1出力の場合はDを指定)	TCP/UDPのボート番号ではない
to: functionKey: "copy-branch-main" port: 0	Connectionに対ける送偏矢 unctionの情報 テータ送偏矢 functionの施財子、Functionsのmapのkey値を設定 テータ送偏矢 functionのカカボート施労番号 (Functionが1.入力の場合は0条指定)	TCP/UDPのボート番号ではない
connectionTypeName: "auto" - from:	ConnectionTypeのリソース名または"auto"を指定 Connectionにおける送偏元Functionの情報	現在の試作は常に"auto"を指定。そのためConnectionTypeカスタムリソースを参照することはない
functionKey: "copy-branch-main" port: 0 to:	データ送根デFunctionの無別子、Functionsのmapのkey値を設定 データ送根デFunctionの出力ボート臨身番号(Functionが1出力の場合は0を指定) Connectionにおける送保だFunctionの情報	TCP/UDPのボート番号ではない
functionKey: "infer-1" port: 0	データ法債子(Functionの識別ではLookの)が特別 データ法債子(Functionの識別・Functionのmapのkey値を設定 データ法債子Functionの入力ポート識別番号(Functionが1入力の場合は0を指定)	TCP/UDPのボート番号ではない
connectionTypeName: "auto" - from:	ConnectionTypeのリソース名または"auto"を指定 Connectionにおける送傷元Functionの情報	現在の試作は常に"auto"を指定。そのためConnectionTypeカスタムリソースを参照することはない
functionKey: "copy-branch-main" port: 1		現住の取作は神に duto を指定。ていたが、Connection Typeがスクロジアー人を参加することは使い
to:	データ送偏元Functionの識別子。Functionsのmapのkey値を設定 データ送偏元Functionの出力ボート識別番号(Functionが1出力の場合はDを指定)	情報がJEPDAM、自由ので指点をいいたCommitted types/プロジップ人を参考をよるより。 TCP/UDPOが下入機能ではない
to: functionKey: "infer-2" port: 0	データ返車Functionの流路手、Functionsのmaptiker僧能定 データ返車Functionの出力ホール監督器号(Functionが)出力の場合はDを指定) Connectionに記さる返車Functionの清略 データ返車Functionの流路手、Functionsのmapoker僧を設定 データ返車Functionの入力ボート監督を呼「Functionが]入力の場合はDを指定)	TCP/LDPのボート番号ではない
functionKey: "infer-2" port: 0 connectionTypeName: "auto" - from:	データ展子putchenの開発デル putchendingのRese 情報音 データ展示putchen Tein + Rese	TCP/LIDPOSE-下面号では51.
functionKey: "infer-2" port: 0 connectionTypeName: "auto"	データ機能予加ctionの開発デールnctionのmankee・情報設定 データ機能デルロに対しが上中機能開発性に同じ的対し対象性は改善設立 Connectionと対しが表現を表現であった。 で一分機能予加ctionの開発・データ機能を開発を データ機能予加ctionの入力が一体開発性であった。 データ機能予加ctionの入力が一体開発性であった。 Connection)を対してみるまとす。本のでは、	TCP/UDPOボー指導ではない TCP/UDPOボー指導ではない TCP/UDPOボー指導ではない 程信のは作り返れて、Add を指定。そのためConnectionTypeカスタムリソースを参照することはない TCP/UDPOボー指導ではおい
functionKey: "infer-2" port: 0 connectionTypeName: "auto" - from: functionKey: "infer-1" port: 0	データ機能子unctionの機能学 F. Austrone/map@ke-機能設定 データ機能子unctionが表示一条機能等性ではいけれたが発生 このmeticin、おする研究チェルではのが構築 データ機能子unctionが表現 データ機能子unctionが表現を発生 データ機能子unctionが、大力を一般能能等化ではていない。 インの機能分れないのが、大力を一般能能等化ではでいない。 インのmeticin、たまかを表現でいないのが発展 データ機能子unctionの(選挙子 F. Austrone/map@ke-機能設定 データ機能子unctionの(選挙子 F. Austrone/map@ke-機能設定 データ機能子unctionの(選挙子 F. Austrone/map@ke-機能設定 データ機能子unctionの(選挙子 F. Austrone/map@ke-機能設定 データ機能子unctionの(選挙子 F. Austrone/map@ke-Map@ke- を発展する。	TCP/UDPOM-ト海号ではない  TCP/UDPOM-ト海号ではない  場合の域内が立流で、adu を指定、そのためConnectionTypeがスタム/リソースを参照することはない  TCP/UDPOM-ト海号ではない  TCP/UDPOM-ト海号ではない  COnnectionTolfでの終了で、(必要無差を受験するが扱アプリ他)の場合は、"wh-end-of-chain" から数さなアチ外を設定すること  「分解析とが発生となったにその体化」だってが開発的を場合は、"wh-end-of-chain" から数さなアチ外を設定すること  「分解析とが発生とれてにその体化」だってが開発的を場合は、"wh-end-of-chain" から数さなアチャル設定すること  「分解析とが発生とれてにその体化」だってが開発を影合は、"wh-end-of-chain" の他に影響や文字列を持ちて、このたのヴィニー・カンステ列を設定す
functionKey: "ufer-2" port: 0 connectionTypeRame: "auto"  LinctionKey: "ufer-1" port: 0 to:  functionKey: "wb-end-of-chain-1"  port: 0 port: 0 functionKey: "wb-end-of-chain-1"	データ機能デルロイビのの選択デー、FunctionのmapのReve可能設定 データ機能デルロイビルが、一般調整・デルロイビのが対象が開発 とのmeteionに対する機能デルロイビのが開発 データ機能デルロイビのの表す。 データ機能デルロイビの関係を一般調整を「functionが1人力が増加される主文 でのmeteion たが、デースをおよす。からで、 のmeteionに対するが、デースをおよす。からで、 で、 データ機能デルロイビのが開発 データ機能があるといるがあるがあるがあるがあるがあるがあるがあるがあるがあるがあるがあるがあるがある	TCP/UDPのボー海骨ではない 現在の域が中で、一部骨ではない 現在の域が中で、一部骨ではない 現在の域が中で、一部骨ではない TCP/UDPのボー海骨ではない TCP/UDPのボー海骨ではない Connectionfloffでのます。(必要は果実を属するがボアリ州自治の場合は、"wb-end-of-chain"が終まる文字外を設定すると が表化とが得るされたにて必ずない。 「************************************
functionKey: "infer-2" port: 0 connectionTypePalame: "auto" - from: port: 0 port: 0 to: to: functionKey: "wb-end-of-chain-1" port: 0 connectionTypePalame: "auto" - from:	データ機能デルnctionの開発デル Functionのinnapiliteの情報記憶 データ機能デルnctionの開発デルをTunction if Internation if Int	TCP/UDPOR-ト語号ではない TCP/UDPOR-ト語号ではない 研究の場合ではない。 研究の場合ではない。 研究の場合ではない。 TCP/UDPOR-ト語号ではない。 TCP/UDPOR-NEW ではない。 TCP/UDPOR-NEW ではないない。 TCP/UDPOR-NEW ではないない。 TCP/UDPOR-NEW ではないないないないないないないないないないないないないないないないないないない
functionKey: "infer-2" port-0 connectionTypePalame: "auto" - from: - f	データ機能デルスの内の開発デルスではTonのProsp Cikeの単位で表現 データ機能デルスでは、対する機能デルスでは、のでは、対しているでは、対し、対しているでは、対しない、対しているでは、対しているでは、対しい、対しない、対しない、対しない、対しない、対しない、対しない、対しない、	TCP/UDPのボー海骨ではない 現在の域が中面で、大きなでは、 現在の域が中面で、大きなでは、 であって、 現在の域が中面で、大きなではない  TCP/UDPのボー海骨ではない  TCP/UDPのボー海骨ではない  Connection/To/FCのボースので、 では、 では、 では、 でいるでは、 でいるではないるでは、 でいるでは、 でいるで
functionKey: "infer-2" port: 0 connectionTypeName: "auto" - from: functionKey: "infer-1" port: 0 port: 0 port: 0 connectionTypeName: "auto" - from: functionKey: "wb-end-of-chain-1"	データ機能デルロの自然開発。Functionの指導を発生 データ機能デルロの記述が一条開催等を行っている対象が開催 データ機能デルロの記述が一般開催を データ機能デルロの記録を、自然に対象を開催を データ機能デルロの記録を、自然に対象を開催を データ機能デルロの記録を、自然に対象を開催している。 データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述が、一般開催性でいるとの記述を データ機能デルロの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でつるとのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとの記述を でのことになっているとのになっている。	TCP/UDPのボート番号ではない TCP/UDPのボート番号ではない 展定の延行はアニュムでを変え、そのためConnectionTypeth2/94/07~入を参修するとはない TCP/UDPのボート番号ではない TCP/UDPのボート番号ではない ConnectionTof/COSFで、企成機能操化を確するが終アが指言が開きた。「から end-of-chain*から指言なアデ列を設定すること 少多体がようと特別とないたに不必体には「大面が開発ある場合は、いら end-of-chain*の機に直接でステデザを対すらない。このためてコニークな文学所を設定する。 実施化・シーラルとれたに不必体には「大面が開発ある場合は、いら end-of-chain*の場合はないでから end-of-chain・yyy? 実施化・シーラルとれたに不必体には「大面が開発ある場合は、いら end-of-chain*などでから end-of-chain・yyy? 実施化・シーラルとはない。 現在の場合はおいまして、では、では、日本では、日本では、日本では、日本では、日本では、日本では、日本で
functionKey: "ufer-2" port: 0 connectionTypeName: "auto" connectionTypeName: "auto" functionKey: "ufer-1" port: 0 to:  functionKey: "wb-end-of-chain-1"  port: 0 connectionTypeName: "auto" - from: port: 0 connectionTypeName: "auto" - functionKey: "wb-end-of-chain-2" - port: 0	データ機能デルロのはの機能学、Functionの指数を保証を データ機能デルロのは対比が一角機能量性ではいていませんが発生 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロののは対した。 データ機能デルロののは対した。 データ機能デルロののは対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロのののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対しためた。 データをに対しためた。 データをに対しためためた。 データをに対しためためためためためためためためためためためためためためためためためためため	TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい  - ConnectionのTofでのはする(後期後単を提昇するが終アプリ担当)の場合は、"wb-end-of-chain" から起きる文学界を設定すること - ***がまたが、**/神参えと打たて深る地に終するが構築を必要は、"wb-end-of-chain" の底に整御文学界を持ちして、このでいてユニークな文学界を設定する - ***(関す: "wb-end-of-chain" の底に付けられた。 ***(アリー・ロールーのでは、 ***) で、 ***(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 ***) で、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 ***) で、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 *
functionKey: "ufer-2" port: 0 connectionTypeName: "auto"  TunctionKey: "ufer-1" port: 0 functionKey: "whe-end-of-chain-1"  port: 0 connectionTypeName: "auto" - from: - from: - from: - from: - from: - functionKey: "whe-end-of-chain-2" - port: 0 - from: -	データ機能子のActionの開発子、ActionのinnapReeの情報記憶 データ機能子がActionの対抗が一般開発子をActionのinnapReeの情報記憶 Connection、おける機能子のActionのinna データ機能子がActionのinnapReeの情報と データ機能子がActionのinnapReeの情報と データ機能子のActionのinnapReeの情報と データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子のActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報記憶 データ機能子がActionのinnapReeの情報	TCP/UDPのボー海号ではは、 TCP/UDPのボー海号ではは、  展示のは作出流に"auto"を指定、そのためConnectionType的スタがリースを参照することはない  CCOnnectionのToがFCのは「な(機能操用を提すする)のボフル側が向着点、"wh-end-of-chain"から終する文字が利息ですること、  "分別ないうとが多えんだけて必ずな。」は、下広が関係なる場合は、"wh-end-of-chain"の高のは、アカリーのでは、  "のないでは、「おいった」では、「ないった」
functionKey, "infer-2" port: 0 connectionTypeName: "auto"  functionKey; "infer-1" port: 0 to:  functionKey; "whe-end-of-chain-1"  port: 0 connectionTypeName: "auto" - from: functionKey; "infer-2" port: 0 for: functionKey; "infer-2" port: 0 functionKey; "infer-2" port: 0 functionKey; "infer-2" functionKey; "infer-2" functionKey; "infer-2" functionKey; "infer-2" functionKey; "and-of-chain-2"  functionKey; "whe-end-of-chain-2"  functionType YaML spayerson: coangot comyvi	データ機能デルロのはの機能学、Functionの指数を保証を データ機能デルロのは対比が一角機能量性ではいていませんが発生 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロのは対した。 データ機能デルロののは対した。 データ機能デルロののは対した。 データ機能デルロののは対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののがは、 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能デルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロのののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学ルロののに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データ機能学のに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対した。 データをに対しためた。 データをに対しためた。 データをに対しためためた。 データをに対しためためためためためためためためためためためためためためためためためためため	TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい TCP/UDPのボー海明でははい  - ConnectionのTofでのはする(後期後単を提昇するが終アプリ担当)の場合は、"wb-end-of-chain" から起きる文学界を設定すること - ***がまたが、**/神参えと打たて深る地に終するが構築を必要は、"wb-end-of-chain" の底に整御文学界を持ちして、このでいてユニークな文学界を設定する - ***(関す: "wb-end-of-chain" の底に付けられた。 ***(アリー・ロールーのでは、 ***) で、 ***(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 ***) で、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 ***) で、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 **(アリー・ロールーのでは、 ***(アリー・ロールーのでは、 *
function(key, "urfer-2" port: 0 connection?pellame: "auto" connection?pellame: "auto" function(key: "urfer-1" port: 0 to:  function(key: "wb-end-of-chain-1"  port: 0 connection?pellame: "auto" - flown: port: 0 connection?pellame: "auto" - function(key: "wb-end-of-chain-2"  port: 0 connection?pellame: "auto" - function?yellame: "auto" - mediadias:   maintipe-decode	データ機能子のLandsのの開発子、Landsonemapticeの情報記憶 データ機能学がLandsの対象が一般開発子がLandsonemapticeの情報と 「Connection」と対象を開発力がLandsonemapticeの情報と データ機能学がLandsonemapticeの情報 データ機能学がLandsonemapticeのは対象力が開発 データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象が開発と データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのが開発が データ機能学がLandsonemapticeのが開発を開発と データ機能学がLandsonemapticeのが表現を Connectionに対象がデータを表現を データ機能学がLandsonemapticeのが表力が開発と データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのである。 データを表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を	TCP/UDPのボー海号ではは、 TCP/UDPのボー海号ではは、  展示のは作出流に"auto"を指定、そのためConnectionType的スタがリースを参照することはない  CCOnnectionのToがFCのは「な(機能操用を提すする)のボフル側が向着点、"wh-end-of-chain"から終する文字が利息ですること、  "分別ないうとが多えんだけて必ずな。」は、下広が関係なる場合は、"wh-end-of-chain"の高のは、アカリーのでは、  "のないでは、「おいった」では、「ないった」
function(key, "ufler-2" port: 0 connection? pediame: "auto" function(key: "ufler-1" port: 0 to:  function(key: "wb-end-of-chain-1" port: 0 connection? pediame: "auto" - from: port: 0 connection? yellame: "auto" - port: 0 connection? yel	データ機能デルロのでの機能学、Functionの作用の保険を検定を データ機能デルロの対比が、一般制度 特別ではいいが、1980年を検定と プータ機能デルロの対象が、1980年を検察を使用しているのが、1980年を検定と データ機能デルロの対象が、1980年を検察を使用しているのが発 データ機能デルロのでは、1980年を使用しているのが発 データ機能デルロのの機能デルロののが開催 データ機能デルロのの機能デルロののが開催 データ機能デルロのの機能デルロののが指 データ機能デルロののが、1980年を使用しているのが発 データ機能デルロののの機能デルロのが自然を使用しているのが表 データ機能デルロののの機能デルロのではのが自然を使用しているのが表 データ機能デルロののの機能デルロののが表 データ機能デルロのの機能デルロののが表 データ機能デルロのの機能デルロののであるのを使用しませを データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デルロのの機能 データ機能デルロのの機能デーチルのが自然を データ機能デルロのの機能データを使用したが可能 データ機能デルロのの機能データを使用したが可能 データ機能デルロのの機能データを使用したが可能 データ機能デルロのの機能データを使用したが可能 データ機能デルロのの機能 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能デルロののが表 データ機能が表 データ機能デルロののが表 データ機能が表 データを表 ・データを表 ・データを表 ・データを表 ・データを表 ・データを表 ・データを表 ・データを表 ・デーを表 ・データを ・データを ・データを ・データを ・データを ・データを ・データを ・データを ・データを ・データを	TCP/UDPのボー海号ではは、 TCP/UDPのボー海号ではは、  展示のは作出流に"auto"を指定、そのためConnectionType的スタがリースを参照することはない  CCOnnectionのToがFCのは「な(機能操用を提すする)のボフル側が向着点、"wh-end-of-chain"から終する文字が利息ですること、  "分別ないうとが多えんだけて必ずな。」は、下広が関係なる場合は、"wh-end-of-chain"の高のは、アカリーのでは、  "のないでは、「おいった」では、「ないった」
functionKey, "wb-end-of-chain-2" port: 0 connectionTypePalame: "auto" - from: port: 0 port: 0 port: 0 port: 0 port: 0 port: 0 functionKey; "wb-end-of-chain-1"  port: 0 connectionTypePalame: "auto" - from: functionKey; "wb-end-of-chain-2" port: 0 to:  functionKey; "wb-end-of-chain-2" port: 0 po	データ機能子のLandsのの開発子、Landsonemapticeの情報記憶 データ機能学がLandsの対象が一般開発子がLandsonemapticeの情報と 「Connection」と対象を開発力がLandsonemapticeの情報と データ機能学がLandsonemapticeの情報 データ機能学がLandsonemapticeのは対象力が開発 データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象力が開発と データ機能学がLandsonemapticeのは対象が開発と データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのが開発が データ機能学がLandsonemapticeのが開発を開発と データ機能学がLandsonemapticeのが表現を Connectionに対象がデータを表現を データ機能学がLandsonemapticeのが表力が開発と データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのである。 データ機能学がLandsonemapticeのが開発 データ機能学がLandsonemapticeのである。 データを表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を	TCP/UDPのボー海号ではは、 TCP/UDPのボー海号ではは、  展示のは作出流に"auto"を指定、そのためConnectionType的スタがリースを参照することはない  CCOnnectionのToがFCのは「な(機能操用を提すする)のボフル側が向着点、"wh-end-of-chain"から終する文字が利息ですること、  "分別ないうとが多えんだけて必ずな。」は、下広が関係なる場合は、"wh-end-of-chain"の高のは、アカリーのでは、  "のないでは、「おいった」では、「ないった」
functionties; "infer-2" port: 0 connectionTypeBalane: "auto" functionties; "infer-1" port: 0 functionties; "infer-1" to:  functionties; "wb-end-of-chain-1"  port: 0 connectionTypeBalane: "auto" - Brom: Rockies; "wb-end-of-chain-1"  port: 0 connectionTypeBalane: "auto" - Brom: Rockies; "wb-end-of-chain-2"  port: 0 to:  functionties; "wb-end-of-chain-2"  port: 0 connectionTypeBalane: "auto" - Brom: Autoconties; "wb-end-of-chain-2"  port: 0 connectionTypeBalane: "auto" - Brom: Autoconties; "wb-end-of-chain-2"  port: 0 connectionTypeBalane: "auto" - FunctionType YAML applevesion: example comy/st todd: TructionType - T	データ機能デルロののの開発デールのはいめの中央の配金機能度 データ機能デルロの対比が一角機能量性ではいめが出た。 Connection、おける機能デルロはのが開発 データ機能が出る。 「一般を受ける。 Connection、おける機能デルロはのが開発 データ機能デルロのでは、 Connection、おける機能デルロはのが開発 データ機能デルロのの開発デールので、機能 データ機能デルロのの開発デールので、機能 データ機能デルロのの開発デールので、 「一般を使ける。 「一般を使ける。 「一般を使ける。」 「一般を使ける。 「一般を使ける。」 「一般を使ける。 「一般を使ける。」 「一般	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functionKey, "ufer-2" port: 0 connectionTypeRiame: "auto"  EuroconKey; "ufer-1" port: 0 functionKey; "ufer-1" to:  functionKey; "wh-end-of-chain-1"  port: 0 connectionTypeRiame: "auto" - from: functionKey; "ufer-2" port: 0 to:  functionKey; "ufer-2" port: 0 port: 0 functionKey; "ufer-2" port: 0 functionKey; "ufer-de-de-dain-2"  FunctionType YAML self-functionType YAML self-functionType yell-meredatata: name: functionType mediatata: name: functionConfiger funct	データ機能力がLinkTonの開発学、FunctionのImageReeの情報記 データ機能力がLinkTonの開発・AutorionのImageReeの情報と データ機能力がLinkTonの開発・AutorionのImageReeの情報を データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報記 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報記 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報 データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を データ機能力がLinkTonのImageReeの情報を フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザ性能に設定する フェーザを ファーゲンタンと フェーザを フェー	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
function(key, "urfer-2" port: 0 connectionTypeBalanne: "auto" connectionTypeBalanne: "auto" function(key: "urfer-1" port: 0 function(key: "urfer-1" to:  function(key: "wb-end-of-chain-1"  port: 0 connectionTypeBalanne: "auto" - Brom: port: 0 port: 0 connectionTypeBalanne: "auto" - Brom: ConnectionTypeBalanne: ConnectionTyp	データ機能デルロののの開発デールのはいめの中央の配金機能度 データ機能デルロの対比が一角機能量性ではいめが出た。 Connection、おける機能デルロはのが開発 データ機能が出る。 「一般を受ける。 Connection、おける機能デルロはのが開発 データ機能デルロのでは、 Connection、おける機能デルロはのが開発 データ機能デルロのの開発デールので、機能 データ機能デルロのの開発デールので、機能 データ機能デルロのの開発デールので、 「一般を使ける。 「一般を使ける。 「一般を使ける。」 「一般を使ける。 「一般を使ける。」 「一般を使ける。 「一般を使ける。」 「一般	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functionKey: "urfer-2" port: 0 connectionTypeName: "auto"  TunctionKey: "urfer-1" port: 0 functionKey: "urfer-1" to:  functionKey: "wb-end-of-chain-1"  port: 0 connectionTypeName: "auto" - from: functionKey: "wb-end-of-chain-2" port: 0 connectionTypeName: "auto" - from: functionKey: "urfer-2" port: 0 for: functionKey: "urfer-2" port: 0 connectionTypeName: "auto" - from: functionKey: "wb-end-of-chain-2" port: 0 connectionType YAML applies comyvt lavid: "functionType yexAmL applies comyvt - functionType - functio	データ機能デルスのでの開発デースのはToneSmapReeの情報記憶 データ機能デルスのは対比が、中級制度 特別ではいいけれらりではなりを担じ Connection、おける機能デルスのはの開発 データ機能デルスのは一般では、 Connection、おける機能デルスのはの開発 データ機能デルスのは、 「アータルをデルスのは、 データ機能デルスのはの開発デースのはToneSmapReeの情報記憶 データ機能デルスのはの開発デースのはToneSmapReeの情報記憶 データ機能デルスのはの開発デースのはToneSmapReeの情報記憶 データ機能デルスのはの開発デースのはToneSmapReeの情報記憶 データ機能デルスのはの開発デースのはToneSmapReeの情報記憶 データ機能デルスののの開発デースのはToneSmapReeの情報記憶 データ機能デルスののの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのはToneSmapReeの情報記憶 データ機能デルスのの開発デースのではToneSmapReeの情報記憶を開発 データ機能デルスのの形成デースのではToneSmapReeの情報記憶を開発 データ機能デルスのの形成形であるの形成を開発 データ機能をデルスのの形成・下のではToneSmapReeの情報記憶を開発 エータが開催を設定する ユーザが作像に設定する ユーザが作像に設定する ユーザが作像に対するのよりによって、 本にはToneSmapPreeのSmapReeのでは、 ToneConeSmapPreeのSmapReeのでは、 ToneConeSmapPreeのSmapReeのでは、 ToneSmapPreeのSmapPreeのでは、 ToneSmapPreeのSmapPreeのでは、 ToneSmapPreeのSmapPreeのでは、 ToneSmapPreeのSmapPr	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
function(key, "urfer-2" port: 0 connection(TypeRame: "auto"  connection(TypeRame: "auto"  function(key; "urfer-1" port: 0 function(key; "urfer-1" port: 0 connection(TypeRame: "auto" - from: function(key; "urbe-nd-of-chain-1" port: 0 connection(TypeRame: "auto" - from: function(key; "urbe-nd-of-chain-2" port: 0 for function(key; "urbe-nd-of-chain-2" port: 0 connection(TypeRame: "auto" - function(Type YAML - gaverance yaverance yavera	データ機能デルスのでの機能デールではToneProspaceの機能設定 データ機能デルスのは対比が一年機能制度 Hourselの対比が一般に対しませた。 Connection Jatic Reflecture Of 11 からできまった。 「一方を発展デルスのは対します。 「一方を表現デルスのは対しません。 Connection Jatic Reflecture Of 11 からできまった。 「一方を表現デルスのは対しません。 「一方を表現デルスのの機能デールでは対します。 「一方を表現デルスのの機能デールでは対します。 「一方を表現デルスのの機能デールでは対します。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールでは対しまからできまった。 「一方を表現デルスのの機能デールではいるがあるが、 「一方を表現デルスのの機能デールではいるがあるが、 「一方を表現デルスのの機能デールではいるがあるが、 「一方を表現デルスのの機能デールではいるがあるをとできまった。 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある。 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの機能デールではいるがある」 「一方を表現デルスのの人力・一般を影響では、「一方を表現デルスのの人力・一般を影響では、「一方を表現デルスの人力・一般を影響では、「一方を表現デルスの人力・一般を影響では、 「「一方がきまいる」 「一方がきまいる」 「一方	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functiontkey: "urfer-2" port: 0 connectionTypeName: "auto"  functiontkey: "urfer-1" port: 0 functiontkey: "urfer-1" to:  functiontkey: "wb-end-of-chain-1"  port: 0 connectionTypeName: "auto" connectionTypeName: "auto" - from: functiontkey: "urbe-nd-of-chain-2" - from: functiontkey: "urbe-nd-of-chain-2" - from: functiontkey: "wb-end-of-chain-2" - port: 0 connectionTypeName: "auto"  FunctionType YAML - applies comyst - indicated connectionTypeName: "auto"  FunctionType YAML - applies comyst - indicated connectionType YAML - applies comyst - indicated connectionType YAML - applies comyst - indicated connectionType - indicated connectionT	データ機能デルスのConnelの部分ド、FunctionalmapReseを確認さ データ機能デルスのConnelのに対した。中級制度 特別ではいるけまりたり機能など Connelのに対したが、中級制度 特別ではいるけまりたり機能など データ機能デルスのごの部分ド、FunctionalmapRese 機能など 「Connelのに対しているのでは、「Annelのでは、Annelのではなどを放立」 Connelのに対しているのでは、「Annelのでは、Annelのでは、Annelのではなどを放立 Connelのに対しているのでは、Annelのでは	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functionKey, "urfer-2" port: 0 connectionTypeName: "auto"  functionKey; "urfer-1" port: 0 functionKey; "urfer-1" to:  functionKey; "wh-end-of-chain-1"  port: 0 connectionTypeName: "auto" from: functionKey; "urfer-2" port: 0 for: functionKey; "urfer-2" port: 0 functionType YAML popt-selection functionComplex compvt port: 0 functionType functionComplex functionCom	データ機能力が大きないたのの開発す。Functionの作用の自体の関係を設定 データ機能が大きないたが、一般開発性ではいめが、一般開発性ではいめが、計分機能などを定 Connection、対する機能が、中心の関係を対しているのでは、対しでいるのでは、対しでいるのでは、対しでいるでは、対しでいるでは、対しているのでは、対しているでは、対しているのでは、対しでいるでは、対しているのでは、対しでいるでは、対しでは、対しでは、対しでは、対しでは、対しでは、対しでは、対しでは、対し	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functions(ey, "urfer-2" port: 0 connection?pellanne: "auto" connection?pellanne: "auto" to:  functions(ey; "urfer-1" port: 0 to:  functions(ey; "wb-end-of-chain-1"  port: 0 connection?pellanne: "auto" - feoni functions(ey; "urfer-2" - governed of-chain-1" functions(ey; "urfer-2" - governed of-chain-2" - feoni functions(ey; "urfer-2" - governed of-chain-2" - port: 0 connection?pellanne: "auto" - functions(ey; "wb-end-of-chain-2" - port: 0 connection?pellanne: "auto"  FunctionType YAML applieston: care place of the connection of t	データ機能デルスのConnelの部分ド、FunctionalmapReseを確認さ データ機能デルスのConnelのに対した。中級制度 特別ではいるけまりたり機能など Connelのに対したが、中級制度 特別ではいるけまりたり機能など データ機能デルスのごの部分ド、FunctionalmapRese 機能など 「Connelのに対しているのでは、「Annelのでは、Annelのではなどを放立」 Connelのに対しているのでは、「Annelのでは、Annelのでは、Annelのではなどを放立 Connelのに対しているのでは、Annelのでは	TCP/UDPのボー海明ではは、 TCP/UDPのボー海明ではは、  展示の域が中国ボールので構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、このなりを構定、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「全体を表現、そのためConnectionType的スタがリースを参照することはない  「CONNECTIONTのでは、「では、国際経典を受けますのボーブが開から開かまた。」、「やからようというでは、「ないで
functionties, "infe-2" port 0 connectionTypelsame: "auto" Innctionties; "infe-1" port 0 functionties; "infe-1" to:  functionties; "wb-end-of-chain-1"  port 0 connectionTypelsame: "auto" form: functionties; "wb-end-of-chain-1" port 0 connectionTypelsame: "auto" form: functionties; "wb-end-of-chain-2" port 0 form: functionties; "wb-end-of-chain-2" port 0 port 0 connectionTypelsame: "auto" functionties; "wb-end-of-chain-2" port 0 port 0 connectionTypelsame: "auto" functiontype YAML applies automatical automatica	データ機能力が大きないたのの開発す。Functionの作用の自体の関係を設定 データ機能が大きないたが、一般開発性ではいめが、一般開発性ではいめが、計分機能などを定 Connection、対する機能が、中心の関係を対しているのでは、対しでいるのでは、対しでいるのでは、対しでいるでは、対しでいるでは、対しているのでは、対しているでは、対しているのでは、対しでいるでは、対しているのでは、対しでいるでは、対しでは、対しでは、対しでは、対しでは、対しでは、対しでは、対しでは、対し	TCP/UDPのボー海岬ではは、 TCP/UDPのボー海岬ではは、  現在の域が中に、  「ログリロのボー海岬ではは、  現在の域が中に、  「ログリロのボー海岬ではは、  「ログリロのボー海岬ではは、  「ログリロのボー海峡ではは、  「ログリロのボー海峡ではな、  「ログリロのボー海峡ではない。  「ログリロのボール・  「ログリロのが、  「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロのが、 「ログリロの

namespace: wbfunc-imgproc	ユーザが任意に設定する	
spec:	ユーソルエ際に収えする ファンクションカタログにおけるGPU高度推論用FPGAフィルクリサイズのファンクション名	
functionName: filter-resize-high-infer functionInfoCMRef:	FunctionChainのFunctionNameに設定される値	
name: funcinfo-filter-resize-high-infer namespace: wbfunc-imgproc	FunctionNameのファンクラミンが主義されているFunctionInfo(ConfigMap)のmetadata.Nameを指定 FunctionNameのファンクラミンが主義されているFunctionInfo(ConfigMap)のmetadata.Namesoaceを指定 Functionのパージョン、Name+Versionで一般性を影響するために使用	
version: 1.0.0	Functionのパージョン。Name+Versionで一意性を担保するために使用	
apiVersion: example.com/v1		
kind: FunctionType metadata:	107000000	
name: functype-filter-resize-low-infer namespace: wbfunc-imgproc	ユーザが任意に設定する ユーザが任意に設定する	
spec: functionName: filter-resize-low-infer	ファンクションカタログにおけるGPU軽量推論用FPGAフィルタリサイズのファンクション名	
functionInfoCMRef: name: funcinfo-filter-resize-low-infer	FunctionChainのFunctionNameに設定される値  FunctionNameのファンルン・サイナ # + 17 / 1 / 4 FunctionInfo / Confidence / Confide	
name: runcimo-liuer-resize-low-inier namespace: wbfunc-imgproc version: 1.0.0	FunctionNameのファンクションが定義されているFunctionInfo(ConfigMap)のmetadata. Nameを指定 FunctionNameのファンクションが定義されているFunctionInfo(ConfigMap)のmetadata. Namespaceを指定	
version: 1.0.0	Functionのパージョン。Name+Versionで一意性を担保するために使用	
apiVersion: example.com/v1		
kind: FunctionType metadata: name: functype-cpu-filter-resize-high-infer	ユーザが任態に設定する	
namespace: wbfunc-imgproc	ユーザが任態に設定する	
functionName: cpu-filter-resize-high-infer	ファンクションカタログにおけるGPU高度推論用CPUフィルタリサイズのファンクション名	
functionInfoCMRef: name: funcinfo-cpu-filter-resize-high-infer	FunctionChainのFunctionNameに設定される値 FunctionNameのファンクションが主義されているFunctionInfo(ConfigMap)のmetadata.Nameを指定	
namespace: wbfunc-imgproc version: 1.0.0	FunctionNameのファンクションが定義されているFunctionInfo(ConfigMap)のmetadata.Namespaceを指定	
Version: 1.0.0	Functionのパージョン。Name+Versionで一意性を担保するために使用	
apiVersion: example.com/v1 kind: FunctionType		
metadata:	ユーザが任像に設定する	
name: functype-cpu-filter-resize-low-infer namespace: wbfunc-imgproc spec:	ユーザが任態に設定する	
functionName: cpu-filter-resize-low-infer	ファンクションカタログにおけるGPU軽量推論用CPUフィルタリサイズのファンクション名	
functionInfoCMRef: name: funcinfo-cpu-filter-resize-low-infer	FunctionChainのFunctionNameに設定される値 FunctionNameのファンクションが定義されているFunctionInfo(ConfigMap)のmetadata.Nameを指定	
name: runcimo-cpu-inter-resize-low-inter namespace: wbfunc-imgproc version: 1.0.0	FunctionNameのアンウェスが正義されているFunctionInfo(LonigMap)のMetadata. Nameを指定 FunctionNameのアンションナだ義教されているFunctionInfo(ConfigMap)のMetadata. Namespaceを指定 Functionのパージョン、Name+Versionで一意性を記録するために使用	
	(株式の200年7 90mmの10.11)	
apiVersion: example.com/v1 kind: FunctionType		
metadata: name: functype-copy-branch	ユーザが任象に設定する	
name: nunctype-copy-branch namespace: wbfunc-imgproc spec:	ユーザが任意に設定する	
spec: functionName: copy-branch	ファンクションかりログにおけるコピー分岐のファンクション名 FunctionChainのFunctionNameに設定される側	
functionInfoCMRef: name: funcinfo-copy-branch		
name: funcinfo-copy-branch namespace: wbfunc-imgproc version: 1.0.0	FunctionNameのアンシラミンが主義されているFunctionInfo(ConfigMap)のmetadata.Nameを指定 FunctionNameのファンクションが主義されているFunctionInfo(ConfigMap)のmetadata.Namespaceを指定 Functionのパージン、Name+Versionで一般性を記修するために使用	
	#10.5 (1)(2) P. 900000-(0.1)	
apiVersion: example.com/v1 kind: FunctionType		
metadata: name: functype-glue-fdma-to-tcp	ユーザが任意に設定する	
namespace: wbfunc-imgproc spec:	ユーザが任態に設定する	
functionName: glue-fdma-to-tcp	ファンクションカタログにおけるGlueのファンクション名 FunctionChainのFunctionNameに設定される値	
functionInfoCMRef: name: funcinfo-glue-fdma-to-tcp		
namespace: wbfunc-imgproc version: 1.0.0	FunctionNameのアンクラミンが主義されているFunctionInfo(ConfigMap)のmetadata.Nameを指定 FunctionNameのアンクラミンが主義されているFunctionInfo(ConfigMap)のmetadata.Namespaceを指定 Functionのパージョン、Name+Versionで一般性を影像するために使用	
Vicanii 1.0.0	Temperature ( Application of the	
apiVersion: example.com/v1 kind: FunctionType		
metadata: name: functype-high-infer	ユーザが任意に設定する	
namespace: wbfunc-imgproc spec:	ユーザが任義に設定する	
spec: functionName: high-infer	ファンクションカタログにおけるGPU高度推論のファンクション名	
functionName: high-infer functionInfoCMRef:	FunctionChainのFunctionNameに設定される値	
functionName: high-infer functionInfoCMRef: name: functinfo-high-infer namespace: withure-imgoroc	Function Chain のティースタースタースタースタースタースタースタースタースタースタースタースタースター	
functionName: high-infer functionInfoCMRef: name: funcinfo-high-infer	FunctionChainのFunctionNameに設定される値 FunctionNameのファンクションが主義されているFunctionInfo(ConfinMao)のmetadata.Nameを指定	
functionName: high-enfer functionName: high-enfer functionNotCHRRd: name: function-logis-lefer name: f	Function Chain のティースタースタースタースタースタースタースタースタースタースタースタースタースター	
functionName: high-enfer functionInfoCHReft: name: functionFolish-infer namespace: wolunc-improc version: 1.0.0	Function/Sensificunt southers 設定計で級 Americon/Sensificunt southers (記述の場合に対応している。 Rection/Sensificunt southers (記述の表述の表述の表述の表述の表述の表述の表述の表述の表述の表述の表述の表述の表述	
functionName: high-infer functionInfoCHReft: name: functionFoth-infer namespace: wolunc-improc version: 1.0.0  Solversion: example.com/v1 sund: FunctionType metadata: name: function-inper namespace: wildurc-improc versions	Function Chain のティースタースタースタースタースタースタースタースタースタースタースタースタースター	
functionName: high-infer functionName: high-infer functionNotNotNotNotNotNotNotNotNotNotNotNotNotN	Function/bandfunction/kames 設定される機  Function/bandfunction/kames 設定される機  Function/bandfunction/kames 対象が表現している。  Function/bandfunction	
functionName: high-wifer functionIndCNRRRI renner: functionEngli-wifer ranner: functionEngli-wifer	Function/AssetSrunction/Asset 設定される機 Function/AssetSrunction/Asset 設定される機 Function/AssetSrunction/AssetSrunction/Info/Confide/AssitSrunction/AssetSrunctio	
functionName: high-infer functionName: high-infer functionNotNotNotNotNotNotNotNotNotNotNotNotNotN	Function/bandfunction/kames 設定される機  Function/bandfunction/kames 設定される機  Function/bandfunction/kames 対象が表現している。  Function/bandfunction	
functionName: high-enfer functionName: high-enfer ramer function legit-infer ramer gracello legit-infer functionName: low-enfer functio	Purction/samiliar/stockhame 設定者の報 Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) ユーザザイ報に設する ユーザザイ報に設する ユーザザイ報に設する スープリイ報に設する カードリイ報に対する カードリイオを カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイオを カードリイを カードリイオを カードリイオを カードリイオを カードリイを カードリイオを カードリイを カードリイを カードリイオを カードリイを	
function/hame: high-enfer function/hame: high-enfer function/hame: function-lagh-infer names function-lagh-infer names guestion-train-infer names function-train-infer names function-train-infer tind function-train-infer tind function-train-infer tind function-train-infer tind function-train-infer funct	Function/hamiffunction/hamis 設定される様 Function/hamiffunction/hamis 設定される様 Function/hamis アンテンテンデルを表れている Function (info: Confide as) (fundadata. Names 報定 Functionのパーラン、Name + Version に 一般生態度するため、皮膚 タープが作業に設する ユーザ作業に設する ユーザ作業に設する ファンテントがの下がけるGV規模議論のファンテットを Function/hamis アンテンナが青まれているチェルを表でいる。 Function/hamis アンテンナが青まれている Function (info: Confide as) (fundadata. Name 報定 Function/hamis アンテンナが青まれている Function (info: Confide as) (fundadata. Name 報定 Function/hamis アンテンナが青まれている Function (info: Confide as) (fundadata. Name 報定	<b>通考</b>
functionName: high-enfer functionName: high-enfer functionName: function brigh-enfer name: function ripse land: Function ripse land: Function ripse land: function ripse metadata: name: function ripse function rips	Purction/samiliar/stockhame 設定者の報 Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) Austrockhame(アンテンシンを開催されている) ユーザザイ報に設する ユーザザイ報に設する ユーザザイ報に設する スープリイ報に設する カードリイ報に対する カードリイオを カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイ報に対する カードリイオを カードリイを カードリイオを カードリイオを カードリイオを カードリイを カードリイオを カードリイを カードリイを カードリイオを カードリイを	優考
Eurosolatione: high-role Eurosolation E	Function/hamiffunction/hamis 設定される機  Function/hamiffunction/hamis 設定される機  Function/hamiffunction/hamis 設定される場合  Function/hamis ファンシュンドを達えないといった。  Decision/hamis ファンシュンドを達えないといった。  Decision/ハージュントをはないといった。  第111年 日本の人・デオー  Decision/ハージュント Name + Network 、	備考
EunctionName: high-wifer  FunctionName: high-wifer  Inner function-bigh-wifer  Inner function-bigh-wifer  Inner function-bigh-wifer  Inner function-bigh-wifer  Inner function-bigh-wifer  Inner function-function  Inner fun	Function/hamiffunction/hamis 設定される機  Function/hamiffunction/hamis 設定される機  Function/hamiffunction/hamis 設定される機  Function/hamis (アンテン・データを入れている function/hamis (Proceedings Confined Associated Confined Associated Confined Associated Confined Associated Confined Associated Associated Confined Associated Assoc	編考
Enricionalmo: high-rofe Enrici	Function/SandfunctionName 設定される機  FunctionName(プァンプンスンが変更れているAutocontent/ConfineNap)がmediatal Nameを報定  FunctionName(プァンプンスンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  DarctionName(プァンプンスンを変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  DarctionName(プァンプンスンを変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  ユーザが他に設定する  ファンタンスが対象が、Autocontent/ConfineNap)がmediatal Namespaceを報定  FunctionNameが、DarctionName 設定される様  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionName の表である。	備考
EurotsonName: high-wifer  FunctionName: high-wifer  InuntonindocNifest:  remer: Faccion-leigh-wifer  remer: faccion-leigh-wifer  remer: faccion-leigh-wifer  remer: faccion-leigh-wifer  remer: faccion-leigh-wifer  remer: faccion-leigh-wifer  rames: faccion-leigh-wifer  rames: faccion-leigh-wifer  function-lingoco-wifer  function-lingoco-wifer  function-lingoco-wifer  rames: faccion-leigh-wifer  function-lingoco-wifer  rames: faccion-leigh-wifer  rames: faccion-leigh-wifer  rames: faccion-leigh-wifer  rames: faccion-leigh-wifer  remer: faccion-leigh-wifer  r	Function/Amerity notion/Americ 設定される機  Function/Amerity notion/Americ 設定される機  Function/Amerity notion/Amerity	爾考
Enricionalmo: high-rofe Enrici	Function/SandfunctionName 設定される機  FunctionName(プァンプンスンが変更れているAutocontent/ConfineNap)がmediatal Nameを報定  FunctionName(プァンプンスンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  DarctionName(プァンプンスンを変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  DarctionName(プァンプンスンを変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定  ユーザが他に設定する  ファンタンスが対象が、Autocontent/ConfineNap)がmediatal Namespaceを報定  FunctionNameが、DarctionName 設定される様  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionNameが、DarctionName の表である。  RuctionName の表である。	<b>通考</b>
functionName: high-enfer functionName: high-enfer ramer functionName: high-enfer ramer function-high-enfer ramer function-high-enfer ramer function-high-enfer ramer function-high-enfer ramer function-high-enfer ramer functionName: how-infer ramerables. functionName: low-infer ramerables: enfence enfer ramerables: enfence enfere ramerables: enfence enfere ramerables: enfence enfere ramerables: enfence enferee ramerables: enfence enferee debicyabletteram: [	Function/saniffunction/sanis 設定される機  Function/saniffunction/sanis 設定される機  Function/saniffunction/sanis 設定される機  Function/sanis (Part	優考
Eurosolation: high-role Eurosolation: Eurosolation: Eurosolation: Eurosolation: Eurosolation: Eurosolation: Familia-fa	Function/sanffunction/sans 設定される機  Function/sanffunction/sans 設定される機  Function/sanffunction/sans 設定される場  Function/sanffu	優考
Enrosolatinos high-role Eurotositance high-role Euroto	Function/sans/function/sans 設定される様  Function/sans/function/sans 設定される様  Function/sans/function/sans 設定される様  Function/sans/func	優勢
Eurosolation: high-role Eurosolation: Eurosolation: Eurosolation: Eurosolation: Eurosolation: Eurosolation: Familia-fa	Function/sanffunction/sans 設定される機  Function/sanffunction/sans 設定される機  Function/sanffunction/sans 設定される場  Function/sanffu	<b>魔考</b>
EurotsonName: high-wifer FunctionIndCNRRR: remer faccionSealph-wifer remerated remer faccionSealph-wifer functionIndCNRRR: ramer functionSealph-wifer functionIndCNRRR: ramer functionSealph-wifer functionIndCNRRR: ramer functionSealph-wifer remerated reme	Function/same/function/same 設定される機  Autolion/same/function/same 設定される機  Autolion/same/function/same 設定される機  Autolion/same/function/same/func	
Enrosolatinos high-role Eurotositance high-role Euroto	Function/sans/function/sans 設定される様  Function/sans/function/sans 設定される様  Function/sans/function/sans 設定される様  Function/sans/func	機ちばかでは未使用
EnctionName: high-wifer FunctionName: high-wifer Innet functionNotes are Innet function begin-wifer Innet function begin-wifer Innet function begin-wifer Innet function begin-wifer Innet functionNotes Innet	Function/sanffunction/sans 設定される機  のよっているからいます。  「Auxilian Control C	
EnrotionName: high-wife  FunctionIndCNRet  Transcriptor  T	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (2014) Function/hams(アンアン・)が書表れているfunction/hams(ConfigNas) (Invested A Namespace) を設定したのでは、アントン・) Names Network (1914)  ユーザ作業に設定する ユーザ作業に設定する ユーザ作業に設定する ファンタン・カナリアに対けるfunction/hams(の対け、	
EnctionName: high-wife  FunctionIndCNReit  Transmission See See See See See See See See See Se	Function/hamffunction/hame 設定される機  Function/hamffunction/hame 設定される機  Function/hamffunction/hame 設定される場  Function/hamffunction/hame 設定される場  Function/hamffunction/hame 会  Durchonのプレンシンとを表していまった。  - 世界主要者である。  - サール・カー・カー・カー・カー・カー・カー・カー・カー・カー・カー・カー・カー・カー・	
EncisionName: high-wifer FunctionIndicKillist  reme: faccinish-wife-remember  version: 1.0.0  version: 1.0.0  policies in complete  report of the complete  function IndicKillist  name: function low-wifer  function IndicKillist  name: function low-wifer  name: function low-wifer  name: function low-wifer  report of the complete  report of th	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (American Hamse) Function/hamse) Function/hamffunction/hamse) Function/hamffunction/hamse Function/hamse) Function/hamffunction/hamse Function/hamffun	
EnctionName: high-wifer  FunctionName: high-wifer  Transe: Turcino-bugh-wifer  Transe: Turcino-bugh-wifer  Transe: Turcino-bugh-wifer  Transe: Surcino-bugh-wifer  Turcino-divorStreet:  Turcino-divorStreet:  Turcino-divorStreet:  Transe: Surcino-bugh-wifer  Transe: Surcino-bugh-wifer  Transe: Surcino-bugh-wifer  Transe: Surcino-bugh-wifer  Transe: Surcino-decode  T	Function/sanffunction/sane 設定される機  のよってのであれるのデンタンとが変更れている。 「Auxilian Confidence Confid	
EncisionName: high-wifer FunctionIndicKillist  reme: faccinish-wife-remember  version: 1.0.0  version: 1.0.0  policies in complete  report of the complete  function IndicKillist  name: function low-wifer  function IndicKillist  name: function low-wifer  name: function low-wifer  name: function low-wifer  report of the complete  report of th	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (American Hamse) Function/hamse) Function/hamffunction/hamse) Function/hamffunction/hamse Function/hamse) Function/hamffunction/hamse Function/hamffun	
Eurocolathockiller Eurocolathock	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (American Hamse) Function/hamse) Function/hamffunction/hamse) Function/hamffunction/hamse Function/hamse) Function/hamffunction/hamse Function/hamffun	
Eurosolatinos high-rofe Eurosolatinos Eurosolati	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (American Hamse) Function/hamse) Function/hamffunction/hamse) Function/hamffunction/hamse Function/hamse) Function/hamffunction/hamse Function/hamffun	
EurotionName: high-wifer FunctionIndCNRRR Transe: Teacher-high-wife reme: Teacher-high-wife reme: Teacher-high-wife reme: Teacher-high-wife reme: Teacher-high-wife reme: Teacher-high-wife remediate: reme: Teacher-high-wife ramespace: wifurc-imagroc goe: functionName: Iow-wifer functionIndCNRRR rames; function-low-wifer ramespace: wifurc-imagroc wersion: 1.0.0  FunctionIndCNRRR ramespace: wifurc-imagroc wersion: 1.0.0  FunctionIndCNRRR ramespace: wifurc-imagroc wersion: 1.0.0  FunctionIndCNRRR ramespace: wifurc-imagroc dealth solivesions: vi teame: _applversion: vi 'reme: "feect", "regionType": "sevSpether", "outputterdeceType": "sevSpether", "outputterdeceType: "sevSp	Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams 設定される機 Function/hamffunction/hams (American Hamse) Function/hamse) Function/hamffunction/hamse) Function/hamffunction/hamse Function/hamse) Function/hamffunction/hamse Function/hamffun	
EunctionName: high-winer  FunctionIndexSided:  Increase Traceins-high-winer Increase Traceins-high-wine	Function/same/Tuncton/same 設定される機  Auxidion/same/Tunchu/same 設定される機  Auxidion/same/Tunchu/same/Same/Same/Same/Same/Same/Same/Same/S	
EunctionName: high-winer  FunctionIndexSides!  Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Incr	Function/same/Tuncton/same 設定される機 (AutoContinus)のであるがある。	
Enricional No. 19th - view of the Control of	Function/sans/function/sans 設定される場  Function/sans/function/sans 設定される。  Function/sans/function/sans 記述によれている。  Function/sans/f	
EunctionName: high-winer  FunctionIndexSides!  Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Tucknot-high-winer Increase Incr	Function/sans/function/sanse 設定される場  Function/sans/function/sanse 設定される場  Function/sans/function/sanse 設定される場  Function/sanse/function/sanse-function/sans	
Eurotionitame: high-wifer Functionitame: high-wifer Functionitations with the second s	Function/sandfunction/sance 設定される場  Function/sandfunction/sance 設定される場  Function/sandfunction/sance 設定される場  Function/sance アンジン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラ	
EncisionName: high-wifer FunctionIndCNRRRI FunctionIndCNRRI Function IndOuter Indigence version: 1.0.0 selective Indigence version: 1.0.0 selective Indigence version: 1.0.0 selective Indigence version: 1.0.0 selective Indigence version: Indigence Indigence Indigence Indigence Indigence Indigence Version: Indigence V	Function-bandfunction/same 設定される機  のよっているいません。  のないません。  の	
EunctionName: high-wine functionHockNiste:  Innet functionHockNiste: In	Function/sandfunction/sance 設定される場  Function/sandfunction/sance 設定される場  Function/sandfunction/sance 設定される場  Function/sance アンジン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラン・グラ	
Eurosolation High-role Eurosolation High-role Eurosolation High-role Eurosolation High-role Indicate High-ro	FunctionStatement/アンシンが書きまたの名前 FunctionStatement/アンシンが書きまたいる中では、 FunctionStatement/アンシンが書きまれて、必要ないるでは、 FunctionStatement/アンシンが書きまれて、必要ないるでは、 FunctionStatement/アンシンが書きまれて、必要ないるでは、 FunctionStatement/アンシンが書きまれて、多までは、 FunctionStatement/Poly と   - エーナが作業に設すする  - エーナが作業に設すする  - エーナが作業に設すする  - エーナが作業に設すする  - エーナが作業に設すする  - アンシンかががたおけるのでは需要基準のアンシンや  FunctionStatement (Poly Application Special	
EncisionName: high-wifer FunctionIndCNRed:  Transe: Special Self-wifer Tran	FunctionStanding Towards 設定される場  FunctionStanding Towards (Table Control of the	
EncisionName: high-wifer FunctionIndCNRed:  Tenner: Security-New York  version: 1.0.0 unders. emigrace  goe:  functionName: low-infer  functionInforName: low-inforName:	Function/same/Tunctookame 設定される機  Autoflookame/Tyn-2w-2/2ff を行れているいた。  Autoflookame/Tyn-2w-2/2ff を行れているいた。  Autoflookame/Tyn-2w-2/2ff を行れているいた。  Autoflookame/Tyn-2w-2/2ff を行れているいた。  Autoflookame/Tyn-2w-2/2ff を行れているいた。  本書も高するため、  Autoflookame/Tyn-2w-2/2ff を行れているいた。  本書も高するため、  Autoflookame/Tyn-2w-2/2ff を行れているいた。  本書も高するため、  スーザが作業に設する  スーザが作業に設する  スーザが作業に設する  スーザが作業に設する  Autoflookame/Tyn-2w-2/2ff を行れているいた。  Autoflookame/Tyn-2w-2/2ff を行れているが、  Autoflookame/Tyn-2w-2/2ff を行れているののである。  Autoflookame/Tyn-2w-2/2ff を行れているのである。  Autoflookame/Tyn-2w-2/2ff を行れているのである。  Autoflookame/Tyn-2w-2/2ff を受けるといののである。  Autoflookame/Tyn-2w-2/2ff を受けるといのである。  Autoflookame/Tyn-2w-2/2ff を使りまました。	

"specName": "spec1"	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファンクションのスペック情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
)'	ファンクションのスペック情報	
spec: '[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	現在の域件では未使用
"name": "spec1", "minCore": 1,	spec配列の要素を参照するための名前 使用するリソースの最小値	
"maxCore": 1,  "maxDataFlowsBase": 1,  "maxCapacityBase": 20,	使用するリソースの最大値 基本の最大部合DataFlow数(最大搭載WBFunction数)。回路のチャネル教等によって決まる	
"max.apacityBase": 20,  "max.inputNum": 1,  "max.OutputNum":1	基本の最大処理能力([ps] ファンクションの最大人力数 ファンクションの最大出力数	
],	27.27.2.12°090/M4790	
- apiVersion: v1 kind: ConfigMap metadata:		
name: funcinfo-filter-resize-high-infer namespace: wbfunc-imgproc data:		
deployableItems: "[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "item1", "regionType": "alveo",	deployableItems配列の要素を参照するための名前 配備可能な領域権別	
"inputInterfaceType": "dev25gether", "outputInterfaceType": "mem",	上記の <majontype>に配備した場合に使用可能な入力のインターフェース種別 上記の<majontype>に配備した場合に使用可能な出力のインターフェース種別 上記の<majontype>に配備して上記の<majontypeをでは、couputinterfacetype>を使用する場合に、デ 上記の<majontype>に配備して上記の<majontype<majontype>に取り<majontype>に取り<majontype>にあり<majontype>に取り<majontype>に対している<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype>にあり<majontype td="" に<="" にあり<majontype=""><td></td></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype></majontype<majontype></majontype></majontypeをでは、couputinterfacetype></majontype></majontype></majontype>	
"configName": "fpgafunc-config-filter-resize-high-infer",	上記のAregionType>に配備して上記のAmputInterfaceType>と <upputinterfacetype>とを使用する場合に、テプロイに必要な情報の名前 上記のAregionType&gt;に配備して上記のAinputInterfaceType&gt;と<utputinterfacetype>と使用する場合のファ</utputinterfacetype></upputinterfacetype>	
"specName": "spec1" },	ンクションのスペック情報の名前	
{     "name": "item2",     "regionType": "alveo",	deployableItems配列の要素を参照するための名前 配備可能な領域種別	
"inputInterfaceType": "mem",  "outputInterfaceType": "mem",	ISS語で呼吸を対象性が 上記のcregionType>に配備した場合に使用可能な入力のインターフェース機別 上記のcregionType>に配備した場合に使用可能な出力のインターフェース機別	
"configName": "fpgafunc-config-filter-resize-high-infer",	上記の <re>上記の<re>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デプロイに必要な情報の名前</outputinterfacetype></inputinterfacetype></re></re>	
"specName": "spec1"	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファンクションのスペック情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
)	コカンパションのフィック連絡	
spec: "[	ファンクションのスペック情報 以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	現在の試作では未使用
"name": "spec1", "minCore": 1,	spec配列の要素を参照するための名前 使用するリソースの最小値	
"maxCore": 1, "maxDataFlowsBase": 8,	使用するリソースの最大値 基本の最大割合DataFlow数(最大搭載WBFunction数)。 図路のチャネル数等によって決まる	
"maxCapacityBase": 40, "maxInputNum": 1, "maxOutputNum":1	基本の最大処理能力(fps) ファンクションの最大人力数 ファンクションの最大人力数	
"maxOutputNum":1 } ]'	ファンクションの最大出力数	
- apiVersion: v1 kind: ConfigMap metadata:		
name: funcinfo-filter-resize-low-infer namespace: wbfunc-imgproc		
data: deployableItems: "[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "item1", "regionType": "alveo",	deployableItems配列の要素を参照するための名前 配備可能な領域権別	
"inputInterfaceType": "dev25gether", "outputInterfaceType": "mem",	上記の <pre>上記の<pre>regionType&gt;に配備した場合に使用可能な入力のインターフェース種別</pre> 上記の<pre>regionType&gt;に配備した場合に使用可能な出力のインターフェース種別</pre></pre>	
"configName": "fpgafunc-config-filter-resize-low-infer",	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デプロイに必要な情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
"specName": "spec1" },	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファンクションのスペック領領の名前</outputinterfacetype></inputinterfacetype></regiontype>	
"name": "item2",	deployableItems配列の要素を参照するための名前	
"regionType": "alveo", "inputInterfaceType": "mem",	配備可能な領域種別 上記の <regiontype>に配備した場合に使用可能な入力のインターフェース種別</regiontype>	
"outputInterfaceType": "mem",  "configName": "fpgafunc-config-filter-resize-low-infer",	上記の <regiontype>に配備した場合に使用可能な出力のインターフェース種別 上記の<regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デ</outputinterfacetype></inputinterfacetype></regiontype></regiontype>	
"outputInterfaceType": "mem",  "configName": "fpgafunc-config-filter-resize-low-infer",  "specName": "spec1"	上記の-regionTypesに配欄して上記の <inputinterfacetype>と<utputinterfacetype>を使用する場合に、デ プロイに必要な情報の名前 上記の<regiontype>に配欄して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype></utputinterfacetype></inputinterfacetype>	
"configName": "fpgafunc-config-filter-resize-low-infer",	上記の-regionTypesに監修して上記の-reput/interfaceTypesとくoutput/interfaceTypesを使用する場合に、デ プロイに必要が開めた場 上記の-regionTypesに監修して上記の-reput/interfaceTypesとくoutput/interfaceTypesを使用する場合のファ ングションのスペック情報の名前	
"configName": "figaflunc-config-filter-resize-low-infer",     "specName": "spec1"     } } spec: [	上記の-regionTypesに配欄して上記の <inputinterfacetype>と<utputinterfacetype>を使用する場合に、デ プロイに必要な情報の名前 上記の<regiontype>に配欄して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype></utputinterfacetype></inputinterfacetype>	現在の試作では未使用
"configName": "fygafunc-config-filter-resize-low-infer",	上記の-regionTypesに配配して記念へreput interfaceType>とくoutputInterfaceType>を使用する場合に、デ プロイに必要な場場の名前 上記の-regionType>に配慮して記念へinputInterfaceType>と <outputinterfacetype>を使用する場合のファ ングションのスペック時報の名前 ファングションのスペック時報</outputinterfacetype>	現在心試作では未使用
"configName": "figafune-config-filter-resize-low-infer", "spect Name": "spect"  "spec: " "manuare": "spect", "maxCoze": 1, "maxCoze": 8,	上記の-regionTypesに比較して上記の-inquitinefraceTypes との如putinefraceTypesを使用する場合に、デ プロイにの様々的影響を 上記の-regionTypesに影響して上記の-inquitinefraceTypesとその如putinefraceTypesを使用する場合のファ プラションのスペック情報 以下iscy-valuatorSuS appoint/ジェクトを要素とする影響を文字列にした機 spec.起列の需素を繋ぎるための基準 使用するリソースの動き機 使用するリソースの動き機 使用するリソースの動き機 使用するリソースの動き機	現在の試作では未使用
"configName": "figafune-config-filter-resize-low-infer", "specName": "spec1" }  spec: " "manuer": "spec1", "maxCore": 1, "maxCore": 1, "maxCore": 4, "maxCore 5, "maxCapachyBase": 8, "maxCapachyBase": 40, "maxIngramer": 40, "maxIngramer": 41,	上記の-regionTypesに比較して上記の-inquitinefraceTypes との如putinefraceTypesを使用する場合に、デ プロイにの様で開発を開 上記の-regionTypesに比較して上記の-inquitinefraceTypesととの如putinefraceTypesを使用する場合のファ プランタンシンのズベック解制 以下Rey-valuatorSuCeSpondTジェクトを要素とする配列を文字列にした機 spec.EXPの要素を繋ぎるEXPOA等 使用するリソースの配子機 毎年のサンチスの配子機 基本の最大効能が(Ps)	現在の試作では未使用
"configName": "fggafunc-config-filter-resize-low-infer",     "specName": "spec1" } "spec: "[ {     "rame": "spec1",     "manCone": 1,     "maxCone": 1,     "maxCone": 8,     "maxCone": 40,	上記の-regionTypesに比較して上記の-inquitinterfaceTypes とのはputinterfaceTypesを使用する場合に、デ プロイに必要では関わない。 上記の-regionTypesに比較して上記の-inquitinterfaceTypesととのよりはInterfaceTypesを使用する場合のファ クラションのスペック情報 ファンクションのスペック情報 ジアとファンションのスペック情報 ジアとファンションのスペック情報 がTriey-valueDysondプジェクトを要高とする影響をア学内にした様 中心に影響を開催するための名前 使用するリソースの影響 衛用するリソースの影響 電子の個大学展示を関係するに使用するでは、 本来の個大学展示を「(fin)	現在の紹介では未使用
"configName": "figalunc-config-filter-resize-low-infer",  "spec.Name": "spec.1"  }  spec: "{  {     "mane": "spec.1",     "masCore": 1,     "masDutarTovsStaes": 8,     "masCores": 1,     "masCores": 1,     "masCopstNum":	上記の-regionTypesに比較して上記の-inquitinefraceTypes との如putinefraceTypesを使用する場合に、デ プロイにの様で開発を開 上記の-regionTypesに比較して上記の-inquitinefraceTypesととの如putinefraceTypesを使用する場合のファ プランタンシンのズベック解制 以下Rey-valuatorSuCeSpondTジェクトを要素とする配列を文字列にした機 spec.EXPの要素を繋ぎるEXPOA等 使用するリソースの配子機 毎年のサンチスの配子機 基本の最大効能が(Ps)	現在のは作では未使用
"configName": "fggafunc-config-filter-resize-low-infer", "specName": "spec1" }  spec: " {     "rame": "spec1",     "manCone": 1,     "maxCone": 1,     "maxCone": 1,     "maxCone": 1,     "maxCone": 1,     "maxCone": 1,     "maxCones": 1,	上記の-regionTypesに比較して上記の-inquitinefraceTypes との如putinefraceTypesを使用する場合に、デ プロイにの様で開発を開 上記の-regionTypesに比較して上記の-inquitinefraceTypesととの如putinefraceTypesを使用する場合のファ プランタンシンのズベック解制 以下Rey-valuatorSuCeSpondTジェクトを要素とする配列を文字列にした機 spec.EXPの要素を繋ぎるEXPOA等 使用するリソースの配子機 毎年のサンチスの配子機 基本の最大効能が(Ps)	現在の試査では未使用
"configName": "figadune-config-filter-resize-low-infer", "specName": "spec1" }  spec: "  "anee": "spec1", "maCozer": 1, "maCozer": 1, "maCozer": 4, "maCozer": 4, "maCozer": 4, "maCozer 1, "maCozer 1	上記の-regionTypesに比較して上記の-inquitinefraceTypes との如putinefraceTypesを使用する場合に、デ プロイにの様で開発を開 上記の-regionTypesに比較して上記の-inquitinefraceTypesととの如putinefraceTypesを使用する場合のファ プランタンシンのズベック解制 以下Rey-valuatorSuCeSpondTジェクトを要素とする配列を文字列にした機 spec.EXPの要素を繋ぎるEXPOA等 使用するリソースの配子機 毎年のサンチスの配子機 基本の最大効能が(Ps)	現在の試合では未使用
"configName": "fggafunc-config-filter-resize-low-infer", "specName": "spec1" }  "spec: " {	上記のでのgonTypesに比較して上記のComputInterfaceTypesとのJUDINterfaceTypesを使用する場合に、デ プロイに必要でAppesに比較して上記のComputInterfaceTypesとのJUDINterfaceTypesと使用する場合のファ プランクションのスペック情報の名 プアンクションのスペック情報 選びません。 プアンクションのスペック情報 選びまな、プランクを受けるJUDINTERFACETYPES といった機 「世界では、JUDINTERFACETYPES といった名 世界では、JUDINTERFACETYPES といった名 世界では、JUDINTERFACETYPES といった名 選手の選択するリンースの最大地 選手の選択する日本の観光を開発しません。 選手の選択する日本の選手を受けるJUDINTERFACETYPES といった。 選手の選択する日本の選手を受けるJUDINTERFACETYPES といった。 選手の選択する日本の選手を受けるJUDINTERFACETYPES といった。 プアングションの選択なります。 ゴアングションの選択なりまする。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なります。 ゴアングションの選択なりまする。 ゴアングションの選択なりまする。 ゴアングションの選択なりまする。 ゴアングションの変する。 ゴアングションの変する。 ゴアングションの変する。 ゴアングションの変する。 ゴアングションの変する。 ゴアングションの変する。 ゴアングラアングションの変する。 ゴアングラアングターの変する。 ゴアングラアングターの変する。 ゴアングラアングターの変する。 ゴアングラアングターの変する。 ゴアングラアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 ゴアングターの変する。 オアングターの変する。 オアングターの変する。 オアングターの変する。 オアングターの変する。 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、 オアングターのなりを、	現在の試合では未使用
"configName": "figafune-config-filter-resize-low-infer", "specName": "spec1"  "spec": "anme": "spec1", "maCore": 1, "maSopartarbroeBase": 8, "masCaparthposBase": 10, "masParthposBase": 10, "masPart	上記の-regionTypesに比較して上記の-insput interfaceTypes との認知はInterfaceTypesを使用する場合に、デ プロイに必要では同かました器して上記の-insput interfaceTypesとその認知はInterfaceTypesを使用する場合のファ プランクションのスペップ情報の話 プアンクションのスペップ情報の話 ジアンクションのスペップ情報の話 対している。 対している。 対している。 では、おいましたがある話 では、おいましたがある話 では、おいましたがある話 では、おいましたがある。 選邦ではアンタースの表示を 選邦ではアンタースの表示を 選邦ではアンタースの表示を 選邦ではアンタースの表示を 選邦ではアンタースの表示を 選邦ではアンタースの表示を 選邦ではアンタースを 選邦ではアンタースを 選びを、では、おいましたがある。 は、では、おいましたが、では、は、は、は、は、は、は、は、は、は、は、は、は、は、は、は、は、は、は	現在の試作では未使用
"configName": "figafune-config-filter-resize-low-infer", "specName": "spec1" } spec: " "anme": "spec1", "maCore": 1, "maCore": 1, "maCore": 1, "maCore": 1, "maCore": 1, "maCore": 1, "maCore filter-resize-light-res	上記の-regionTypesに比較して上記の-inquitinterfaceTypesとのJuputinterfaceTypesを使用する場合に、デ プロイにの確認的場合語  上記の-regionTypesに比較して上記の-inquitinterfaceTypesとそのJuputinterfaceTypesを使用する場合のファ ンクションのスペック情報の名  ファンクションのスペック情報  ジアンクションのスペック情報  「アンクションのスペック情報  「アンクションのスペック情報  「アンクションのスペック情報  「アンクションのスペック情報  「アンクションのスペック情報  「アンクションの表となった。」  「アンクションの表となった。」  「アンクションの表となった。」  「アンクションの表となった。」  「アンクションの表えたが、 「アンクションの表表を表えたが、 「アンクションの表えたが、	現在の試作では未使用
"configName": "figealunc-config-filter-resize-low-infer", "specName": "spec1" }  Spec: "  "anner": "spec4", "macCode*: 1, "macCo	上記の-region/types-に影響して上記の-input/interface/types-と couput/interface/types-を使用する場合に、デ プロイにの様々的影響を開 上記の-region/types-に影響して上記の-input/interface/types-と couput/interface/types-を使用する場合のファ プレラュンのスケック情報の名前 以下にかっていません。 ファングションのスケック情報 以下にかっていません。 「ファングションのスケック情報 以下にかっていません。 「クリースケック情報 以下にかっていません。 「クリースケック情報 は下にかっていません。 「クリースケック情報 は下にかっていません。 「クリースケック情報 は不らな人が理解がある。」 「関係の事業を開催するためる第 の例するリソースの最大機 基本の最大場合のためで表した。 「クリースケックは、 ファングションの最大性 基本の最大地の自己の大きないません。 「ファングションの最大性 大きないません。」 以下にかっていません。 「ファングションの最大性 大きないません。 「アングションの最大性 大きないません。」 は「New-volucifyらはるjoond プジェクトを表面とする配列を文字列にした機 はないません。 は「New-volucifyらはるjoond プジェクトを表面とする配列を文字列にした機 はないません。 は「New-volucifyらはるjoond プジェクトを表面とする配列を文字列にした機 はないません。 は、これには、「ファング・ファング・ファング・ファング・ファング・ファング・ファング・ファング・	現在の試作では未使用
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1" }  Spec: "  "anne": "spec1", "macConfigName": "spec1", "macConfigName": 18, "macConfigName": 18, "macConfigName": 18, "macConfigName": 19, "specName": "port 1000pether", "configName": "cpufunc-config-filter-resize-high-infer", "specName": "spec1"	上記の-regionTypesに比較して上記の-input/interfaceTypesとその即put/interfaceTypesを使用する場合に、デ プロイにの確認的の影響 上記の-regionTypesに比較して上記の-input/interfaceTypesとその即put/interfaceTypesを使用する場合のファ プランのランのスペップ機能の名割 人工やサットの場合というでは、アファンのランのスペップ機能の名割 人工やサットの場合というでは、アファンのランのスペップ機能の名割 人工やサットの場合というでは、アファンのランのスペップ機能を発 を受けるシンの名のスペップ機能を発 を使用するシンスの必要を使用するとの必要 使用するシンスの必要 使用するシンスの必要 使用するシンスの必要 を使用する場合というでは、アファンのランの名が使えて表現を定 では、アファンのランンの最大出力度 ファンクランンの最大出力度 ファンクランンの最大出力度 しているアンのものよりにでは、アファンのようの最大には、アファンのようの最大出力度 しているアンのようなというでは、アファンのような場合というでは、アファンのようなの場合というでは、アファンのようないのでは、アファンのでは	
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1"  }  Spec: "  "manuface": " "manuface	上記の-regionTypesに関連して上記の-input/interfaceTypes との認知はInterfaceTypesを使用する場合に、デ フロイにの確認的場合第  上記の-regionTypesに関連して上記の-input/interfaceTypesとその即はInterfaceTypesを使用する場合のファ ンクションのスペック情報 メアルファンのスペック情報 メアルファンのスペック情報 メアルファンのスペック情報 メアルファンのスペック情報 メアルファンのスペック情報 メアルファンのションのスペック情報 メアルファンのションのスペック情報  オアルファンのションのスペック情報  オアルファンのションのスペック情報  オアルファンのションの表別を発生する配列を文字列にした機  神術するソンースの品を呼吸  オルスの場合	現在の試作では未使用
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1" }  Spec: "{  *aname": "spec4", "maxCare": 1, "maxCare": 1, "maxCare": 1, "maxCare to 1, "maxC	上記の-regionTypesに関連して上記の-inputInterfaceTypes との認知はInterfaceTypesを使用する場合に、デ プロイにの確認的場合の第 上記の-regionTypesに関連して上記の-inputInterfaceTypesとそのUputInterfaceTypesを使用する場合のファ プランタンのスペック情報 以下は中では、アランタンスペック情報 以下は中では、アランタンスペック情報 以下は中では、アランタンスペック情報 以下は中では、アランタンタンの表現を発生する配列を文字列にした機 では、アランタンタンの最近機 最初するリンタンの最近地様 最初するリンタンの最近地様 最初するリンタンの最近地様 最初するリンタンの最近地様 最初するリンタンの最近地様 最初するリンタンの最近地様 最初するリンタンの最近地様 最初ないまでは、アランタンの最近地様 最初ないませたが、アランタンの最近には、アランタンの最近には、アランタンタンのの最大力放 ファンタンタンの最大力放 ファンタンタンの最大力放 ファンタンタンの最大力放 ファンタンタンの最大力放 ファンタンタンの最大力放 ファンタンタンの最大力放 ファンタンタンのの最大力放 ファンタンタンのの最大力放 エ記の-regionTypesに定慮して見合いた例がでは、アランタンスを使用する場合に、デ プロイにの様々が関心を指 上記の-regionTypesに定慮して記念で、中間では、2世ので、アランターを使用する場合に、デ プロイにの様々的なの指 上記の-regionTypesに定慮して上記の-reputInterfaceTypesとを使用する場合に、デ プレイにの様々的地位を指 上記の-regionTypesに定慮して上記の-reputInterfaceTypesとを使用する場合のファ プランタースのスペック情報 以下は中では、アランターを表現とする配列を文字列にした機 マpecと数で発展を整理するための名様	
"configName": "fignature-config-filter-resize-low-infer", "spectName": "spect"  "spect"  "anne": "spect", "macCore": 1, "spect", "spect", "spect 1, "spect", "spect", "spect 1, "spect": "spect", "spect": "spect", "spect": "spect", "macCore": ", "macCore": 1, "macCore": 1	上記の-regionTypesに比較して上記の-input/interfaceTypesとその即put/interfaceTypesを使用する場合に、デフロイにの確認的場合の音 上記の-regionTypesに比較して上記の-input/interfaceTypesとそのuput/interfaceTypesを使用する場合のファンクランのスペック情報 上記の-regionTypesに対象して上記の-input/interfaceTypesとそのuput/interfaceTypesを使用する場合のファンクランのスペック情報 上では、またが、またが、またが、またが、またが、またが、またが、またが、またが、またが	
"configName": "figadum-config-filter-resize-low-infer", "spect Name": "spect 1" " "spect " "anme": "spect 1", "maxCore": 1, "maxCore": 1, "maxCore": 1, "maxCore 1	上記の-regionTypesに比較して上記の-input interfaceTypes との如此はinterfaceTypesを使用する場合に、デ  プロイにの様々の場合を開  上記の-regionTypesに関係して上記の-input interfaceTypesとその如此interfaceTypesを使用する場合のファ  プランランのスペック解  以下top-valuatorSuc Spond プラェクトを要素とする配列を文字列にした機  ないました。 「クランカンのスペック解  以下top-valuatorSuc Spond プラェクトを要素とする配列を文字列にした機  のまたの影子機  展示るリンカスの思子機  展示るリンカスの思子機  展示るリンカスの思子機  展示を対象を関係するための名称  のまたの表示を表示を表示を表示を表示を表示を表示を表示を表示を表示を表示を表示を表示を表	
"configName": "figadune-config-filter-resize-low-infer", "specName": "spec1"  "spec", "manufacture-til, "masCapachyBase": 40, "masCapachyBase": 50, "masCapachyBase": 60, "special control countries of the country of the count	上記の-regionTypesに比較して上記の-input interfaceTypes と couput interfaceTypes を使用する場合に、デ プロイにの確認的場合の音 上記の-regionTypesに比較して上記の-input interfaceTypesと couput interfaceTypesを使用する場合のファ プランタンのスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムを受けるというというというというというというというというというというというというというと	
"configName": "figadune-config-filter-resize-low-infer", "specName": "spec1" } spec: " "manufacter": 1, "maxCard-1, "regoin-yee": "card-1,	上記の-regionTypesに比較して上記の-input interfaceTypes と couput interfaceTypes を使用する場合に、デ プロイにの確認的場合の音 上記の-regionTypesに比較して上記の-input interfaceTypesと couput interfaceTypesを使用する場合のファ プランタンのスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムをクラックスペック情報 以下をアッカルログラムを受けるというというというというというというというというというというというというというと	
"configName": "fignature-config-filter-resize-low-infer", "spectName": "spect", "spectName": "spect", "farme": "spect", "macCare": 1, "macCare operfilter-resize-high-infer namespace: widhure-insprace data: deployabiliteria: (" "spect": "spect", "spect "spect", "macCare": 1, "	上記の-regionTypesに比較して上記の-reputitierfaceTypesとの認知はinterfaceTypesを使用する場合に、デ 上記の-regionTypesに関連して上記の-reputitierfaceTypesとのの対しては合う。 上記の-regionTypesに関連して上記の-reputitierfaceTypesとの対しているのである。 ファンクションのスペック開製 以下isoy-valuathのはるpondプラエクトを要素とする配列を文字列にした値  specativeの要素を繋ぎるための経過 一般用するリンテスの最大値 選手を受け、一次の最大値 選手を受け、一次の最大値 基本の最大が開発が(Psi) ファンクションのスペック開製 以下isoy-valuathのは、多数では、 基本の最大が開発が(Psi) ファンクションの最大地力度  はTisoy-valuathのは必要が開発がPsi unctionを)。回路のチャネル表等によって決まる 基本の最大が開発が(Psi) ファンクションの最大が放 ファンクションの最大が放 ファンクションの最大が放 と思うできないの対けのないとに関連した場合に使用可能となよりのインターフェース機関 上記の-regionTypesに定極して影響した場合に使用可能となよりのインターフェース機関 上記の-regionTypesに定極して影響に使用で表現のインターフェース機関 上記の-regionTypesに定極して影響に使用で表現のインターフェースを受け上記のよりに関連して記憶した場合に使用で表現のインタンタンのスペック情報の名前 上記の-regionTypesに定極してまたのようを指 で対するマックで表現の名前 はアナンクションのスペック情報の名前 はアナンクションのスペック情報の名前 はアナンクションのスペック情報の名前 はアナンクションのスペック情報の名前 はアナンクションの表が大力技 ファンクションの最大人力技 コアンクションの最大人力技 コアンクションの最大大力技 ファンクションの最大大力技 ファンクションの最大大力技 ファンクションの最大大力技	
"configName": "figadune-config-filter-resize-low-infer", "spect Name": "spect " "spect" "anie": "spect", "macCoard-stand-ordered "spect", "spect ordered "spect", "macCoard-stand-ordered "spect "spect", "macCoard-stand-ordered "spect	上記の-regionTypesに比較して上記の-reputitienfaceTypesとその即putitienfaceTypesを使用する場合に、デ 上記の-regionTypesに関連して上記の-reputitienfaceTypesとそののputitienfaceTypesを使用する場合のファ ンプションのスペック解除 以下key-valuetからならpondプラェクトを要素とする配列を文字列にした値 specativeの要素を繋ぎるための各種 使用するリンースの最大値 基本の最大対象の関係を繋ぎるための名様 を基本の最大対象の内容がである。 以下key-valuetからならpondプラェクトを要素とする配列を文字列にした値 のput まつりースの最大値 基本の最大対象が作り。 ファングションの最大が放 ファングションの最大が放 ファングションの最大が放 ファングションの最大が放 ファングションの最大が放 ファングションの最大が成 は「key-valuetからならpondプラェクトを要素とする配列を文字列にした値 dasployabeltemackがの要素を繋ぎるための名様 に最初では関係制度 上記の-regionTypesに定能した場合に使用可能なよ力のインターフェース機削 上記の-regionTypesに定能した場合に対しまる。 に関係では関係制度 上記の-regionTypesに定能した場合に使用可能なよ力のインターフェース機削 上記の-regionTypesに定能した場合に使用可能なよ力のインターフェース機削 上記の-regionTypesに定能したと思いては同じたを表が表する場合に、デ プロイにの観せ解析の名前 上記の-regionTypesに定能して上を記しているのではではではできないのである。 アンシンコンのスペック外側 の表述 ファングションのスペック外側 の表述 の表示を観音を表示とから系数 のままりとうる表述 本本の最大が観音を表が表が表が、回路のチャネル表等によって決まる 本本の最大が観音を表が表が、自然	
"configName": "figadum-config-filter-resize-low-infer", "specName": "spec1" } spec: "{     "rame": "spec1",     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 8,     "maxCoard-overlage: 8,     "maxCoard-overlage: 8,     "maxCoard-overlage: 8,     "maxCoard-overlage: 8,     "maxCoard-overlage: 9,     "maxCoard-overlage: 9,     "maxCoard-overlage: 9,     "maxCoard-overlage: 9,     "maxCoard-overlage: 9,     "and-overlage: 9,     "overlage: 9,     "configName": "court,     "spectand-overlage: 9,     "spectand-overlage: 9,     "spectand-overlage: 9,     "spectand-overlage: 9,     "maxCoard-overlage: 9,     "maxCoard-overlage: 1,     "m	上記の-regionTypesに比較して上記の-input interfaceTypes との即put interfaceTypesを使用する場合に、デ 上記の-regionTypesに関係して上記の-input interfaceTypesとを使用する場合のファ ンクションのスペック解除 以下key-valuetからならpondプラェクトを要素とする配列を文字列にした値  spec 起がの要素を繋ぎるための各種 使用するリンースの扱う値 使用するリンースの扱う値 使用するリンースの最大値 基本の最大地を関するための名様 は下key-valuetからならpondプラェクトを要素とする配列を文字列にした値  app は	
"configName": "fignature-config-filter-resize-low-infer", "spectName": "spect", "spectName": "spect", "mare: "spect", "marcare: "spect", "sport/spect", "sport/spect", "spect", "spect", "spect", "spect", "spect", "marcare: "spect", "marcare: "spect", "spect", "marcare: "spect", "	上記の-regionTypesに関連して上記の-input interfaceTypes との如put interfaceTypesを使用する場合に、デ  プロイにの様での場合を開  上記の-regionTypesに関連して上記の-input interfaceTypesととの如put interfaceTypesを使用する場合のファ  プランタンタンのスペック開  以下をヴァンタンタンのスペック開  以下をヴァンタンタンのスペック開  以下をヴァンタンタンのスペック開  以下をヴァンタンタンのスペック開  以下をヴァンタンタンのスペック開  以下をヴァンタンタンのスペック開  選手の最大力を関  選手の最大力を関  プランタンタンの最大権  基本の最大効理能が(Psi)  プランタンタンの最大力を  プランタンタンの最大力を  プランタンタンの最大力を  プランタンタンの最大力を  プランタンタンの最大力を  プランタンタンの最大力を  プランタンタンの最大力を  に最初では、は、「大きな、「大きな、「大きな、「大きな、「大きな、「大きな、「大きな、「大きな	
"configName": "fgpafune-config-filter-resize-low-infer", "spectName": "spect", "ane": "spect", "marCore": 1, "marC	上記の-regionTypesに関連して上記の-input interfaceTypes との如put interfaceTypesを使用する場合に、デ  プロイにの様では同じた。 上記の-regionTypesに関連して上記の-input interfaceTypesととの如put interfaceTypesを使用する場合のファ  プランタンションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションのスペック開輸 以下をヴァッションの基本権 基本の最大が開業が作り。 ファンクションの最大地 大阪 はアルタンターの最大地 はアルタンターのまた。 はアルタンターのまた。 はアルタンターのよりに関ルでした。 はアルタンターのよりに関ルでした。 はアルタンターのよりに関ルでした。 はアルタンターのよりは はアルタンのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルタンターのよりは はアルターのよりは はアルタンターのよりは はアルターのよりは	
"configName": "figealunc-config-filter-resize-low-infer", "specName": "spec1"  }  spec: "  {	上記の-responTypesに関係して上記の-input interfaceTypes と edipt する場合に、デ	
"configName": "figadum-config-filter-resize-low-infer", "specName": "spec1" } spec: " {     "rane": "spec1",     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "spect-stand-overlage: 9,     "spect-stand-overlage: 9,     "spect-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 1,     "maxCoard-stand-overlage: 1,	上記の-regionTypesに関連して上記の-reputitiverfaceTypesとその即putitiverfaceTypesを使用する場合に、デ 上記の-regionTypesに関連して上記の-reputitiverfaceTypesとそののputitiverfaceTypesを使用する場合のファンクションのスペック解除 リアンクションのスペック解除 リアンクションの最大性 カアンクションの最大性 カアンクションの最大性 カアンクションの最大性 カアンクションの最大性 大変を使用する場合の表別 リアンクションの最大性 大変を使用する場合の表別 レビルの-regionTypesに関連して単常に使用可能とはカロインターフェース種別 上記の-regionTypesに関連して単常に使用可能となカロインターフェースを使用する場合に、デ ファンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解除 リアンクションのスペック解析の名割 日本で成まれたが表別の表別 日本では、日本では、日本では、日本では、日本では、日本では、日本では、日本では、	
"configName": "figadum-config-filter-resize-low-infer", "spect Name": "spect " "spect" "ranee": "spect", "macCoard-stand-overlage": 8, "macCoard-stand-overlage - 1, "macCoard-stand-overlage - 1, "macCoard-overlage - 1, "macCoard	上記の-respont/psp-に関連して上記の-reputitierface/type> と-couputitierface/type>を使用する場合に、デ 上記の-respont/psp-に関連して上記の-reputitierface/type>とそののputitierface/type>を使用する場合のファ ンクションのスペック解除 以下top-valuatringのでは、アルールを表現を実施した機 コアンクションのスペック解除 以下top-valuatringのでは、アルールを表現を実施した機 コアンクションのスペック解除 以下top-valuatringのでは、アルールを表現を表現を実施した機 関邦をリソースの最大機 基本の最大対象の表現を観響するための各種 関邦をリソースの最大機 基本の最大対象の作り。 ファンクションの最大社 カタファングションの最大社 カタファングションの最大社 大変 は下top-valuatringのでは、アルールを表現を大きが大きな表現を実施した機 は同様の表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表現を表	現在の試合では未使用
"configName": "figadum-config-filter-resize-low-infer", "specName": "spec1" } spec: " {     "rane": "spec1",     "maxCoard-stand-overlage: 8,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "spect-stand-overlage: 9,     "spect-stand-overlage: 9,     "spect-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 9,     "maxCoard-stand-overlage: 1,     "maxCoard-stand-overlage: 1,	上記の-respont/pserンに設成して上記の-resput/interface/type> との認知がinterface/type>を使用する場合に、デ 上記の-respont/pserンに設成して上記の-resput/interface/type>とその即用する場合のファ ンクションのスペック解除 以下top-valuatrinのは、おからのオラジェクトを要素とする配列を文字列にした機 のpsecを持つ要素を繋ぎるための各等 を関するリンテスの最大機 基本の最大対象の対象を関するための名等 を基本の最大対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対	現在の試合では未使用
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1"  "spec" [  "rame": "spec1", "masCoat-state	上記のでsgonTypesに関連して上記のでsputInterfaceTypesとその如putInterfaceTypesを使用する場合に、デ 上記のでsgonTypesに関連して上記のでsputInterfaceTypesとその如putInterfaceTypesを使用する場合のファ ンクションのスペック情報 していまった。 ファンクションのスペック情報 していまった。 ファンクションのスペック情報 していまった。 ファンクションのスペック情報 していまった。 ファンクションのスペック情報 していまった。 ファンクションのスペック情報 していまった。 ファンクションのスペック情報 は下をいった。 ファンクションのスペック情報 選手の意見が要素と思するとかの名前 を持てるリンターの最大力度 ファンクションの最大力度 ファンクションのア・に発しては他に一般の中間をとならかシャク・フェース機能 上記のでsgonTypesに発化しては他に一般の中間をとならかシャク・フェース機能 上記のでsgonTypesに発化しては他に一般の中間をとならかシャク・フェース機能 上記のでsgonTypesに発化しては他に一般の中間をとならかシャク・フェース機能 上記のでの要が下からに発化しては他に一般の中間をとならかシャク・ファンクリーののスペック情報の名前 ファンクションのスペック情報の名前 ファンクションのスペック情報の名前 ファンクションのスペック情報 関邦をリソースの最小機 機能するの場が、表現していまった。 本本の個大学機能の機能を表現していまった。 エドをマールの中でのできた。 エドをマールの中でのできた。 エドをマールの中でのできた。 エドとのでsgonTypesに関心では、自然では同じなといった。 エドとのでsgonTypesに関心では、自然では同じなといった。 エドとのでsgonTypesに関心では、自然では同じなといった。 エドとのでsgonTypesに関心では、自然では同じなといった。 エドとのでsgonTypesに関心では、自然では同じなといった。 エドとのでsgonTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでsponTypesに関心では、正常のでないまからないまからないまからないまからないまからないまからないまからないまから	現在の社会では未使用
"configName": "specis" "spechiame": "specis" "spechiame": "specis" "specis" "specis": "mancare "specis", "specis "specis", "specis "specis", "specis "specis", "specis "specis", "specis "specis", "mancare "specis "specis "specis specis spe	上記のでsponTypesに関連して上記のでsputInterfaceTypesとその即用する場合に、デ コンマンションのスペック解除 コアンクションのスペック解除 コアンクションの表と解せて、またので、またので、またので、またので、またので、またので、またので、またので	現在の試合では未使用

]* - apiVersion: v1		
kind: ConfigMap metadata:		
name: funcinfo-copy-branch name: funcinfo-copy-branch namespace: wbfunc-imgproc		
data:		
deployableItems: "[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "item1",	deployableItems配列の要素を参照するための名前	
"regionType": "cpu", "inputInterfaceType": "host100gether",	配備可能な領域権別 上記の <regiontype>に配備した場合に使用可能な入力のインターフェース種別</regiontype>	
"outputInterfaceType": "host100gether",	上記の <regiontype>に配備した場合に使用可能な出力のインターフェース種別</regiontype>	
"configName": "cpufunc-config-copy-branch",	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デ プロイに必要な情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
"specName": "spec1"	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype>	
spectrame . spect	ンクションのスペック情報の名前	
J'		
spec: '[	ファンクションのスペック情報 以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	現在の試作では未使用
{	spec配列の要素を参照するための名前	
"name": "spec1", "minCore": 1,	spec能列の要素を参照するための名前 使用するリソースの最小値	
"maxCore": 1,	使用するリソースの最大値 基本の最大割合DataFlow数(最大搭載WBFunction数)。 図路のチャネル教等によって決まる	
"maxDataFlowsBase": 1, "maxCapacityBase": 15,	基本の眼大副音DataHow数(眼天拾載WBrunction数)。四路のナヤイル数号によって決まる 基本の最大処理能力(fps)	
"maxInputNum": 1,	ファンクションの最大入力数	
"maxOutputNum":10 }	ファンクションの最大出力数	
) - - apiVersion: v1		
kind: ConfigMap		
metadata: name: funcinfo-glue-fdma-to-tcp		
namespace: wbfunc-imgproc data:		
deployableItems: "[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "item1",	deployableItems配列の要素を参照するための名前	
"regionType": "cpu",	配備可能な領域種別 上記の <regiontype>に配備した場合に使用可能な入力のインターフェース種別</regiontype>	
"inputInterfaceType": "mem", "outputInterfaceType": "host100gether",	上記の <regiontype>に配備した場合に使用可能な人力のインターフェース機別 上記の<regiontype>に配備した場合に使用可能な出力のインターフェース機別</regiontype></regiontype>	
"configName": "cpufunc-config-glue-fdma-to-tcp",	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デ プロイに必要な情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype>	
"specName": "spec1"	ンクションのスペック情報の名前	
)'		
spec: '[	ファンクションのスペック情報 以下key-valueからなるisonオブジェクトを裏蓋とする配列を文字列にした値	現在の試作では未使用
{	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "spec1", "minCore": 1,	spec配列の要素を参照するための名前 使用するリソースの最小値	
"maxCore": 1,	使用するリソースの最大値	
"maxDataFlowsBase": 1,	基本の最大割合DataFlow数(最大搭載WBFunction数)。回路のチャネル数等によって決まる 数まの最大制度的な(fee)	
"maxCapacityBase": 15, "maxInputNum": 1,	基本の最大処理能力(fps) ファンクションの最大入力数	
"maxOutputNum":1	ファンクションの最大出力数	
)'		
- apiVersion: v1 kind: ConfigMap		
metadata: name: funcinfo-high-infer		
namespace: wbfunc-imgproc		
data: deployableItems: "[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
"name": "item1",	deployableItems配列の要素を参照するための名前	
"regionType": "a100",	配備可能な領域種別	
"inputInterfaceType": "host100gether", "outputInterfaceType": "host100gether",	上記の <regiontype>に配備した場合に使用可能な入力のインターフェース種別 上記の<regiontype>に配備した場合に使用可能な出力のインターフェース種別</regiontype></regiontype>	
	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デ</outputinterfacetype></inputinterfacetype></regiontype>	
"configName": "gpufunc-config-high-infer",	プロイに必要な情報の名前 上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype>	
"specName": "spec1"	ンクションのスペック情報の名前	
},		
"name": "item1",	deployableItems配列の要素を参照するための名前	
"regionType": "a100", "inputInterfaceType": "mem",	配備可能な領域種別 上記の <regiontype>に配備した場合に使用可能な入力のインターフェース種別</regiontype>	
"outputInterfaceType": "host100gether",	上記の <regiontype>に配備した場合に使用可能な出力のインターフェース種別</regiontype>	
"configName": "gpufunc-config-high-infer",	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合に、デ プロイに必要な情報の名前</outputinterfacetype></inputinterfacetype></regiontype>	
"specName": "spec1"	上記の <regiontype>に配備して上記の<inputinterfacetype>と<outputinterfacetype>を使用する場合のファ</outputinterfacetype></inputinterfacetype></regiontype>	
}	ンクションのスペック情報の名前	
J'	ファングションのスペック情報	
spec: '[	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	現在の試作では未使用
"name": "spec1",	spec配列の要素を参照するための名前	
"minCore": 1,	使用するリソースの最小値	
"maxCore": 1, "maxDataFlowsBase": 1,	使用するリソースの最大値 基本の最大割合DataFlow数(最大搭載WBFunction数)。 回路のチャネル数等によって決まる	
"maxCapacityBase": 15,	基本の最大処理能力(fps)	
"maxInputNum": 1, "maxOutputNum": 1	ファンクションの最大入力数 ファンクションの最大出力数	
}		
]* - apiVersion: v1		
kind: ConfigMap metadata:		
name: funcinfo-low-infer		
namespace: wbfunc-imgproc data:		
deployableItems: "[		
deployablestems: [	以下key-valueからなるjsonオブジェクトを要素とする配列を文字列にした値	
{ "name": "item1",	deployableItems配列の要素を参照するための名前	
{	deployableItems配列の要素を参照するための名前 最適可能とは微速観別 上記の-cregionType>に配偏した場合に使用可能な入力のインターフェース種別	
{ "name": "item1", "regionType": "t4",	degloyable/temeds等の要素を参照するための名前 を無可能な領域観測 上記の-regionTypesに影響した場合に使用可能な入力のインターフェース展別 上記の-regionTypesに設備した場合に使用可能な幼力のインターフェース展別	
{   "name": "Rem1",   "regionType": "t4",   "inputInterfaceType": "host100gether",	deployableItems配列の要素を参照するための名前 最適可能とは微速観別 上記の-cregionType>に配偏した場合に使用可能な入力のインターフェース種別	
{ "name": "fem1", "regonTyee": "14", "regonTyee": "14", "sputInterfacetryee": "host100gether", "outputInterfaceType": "host100gether", "corfigName": "goufunc-config-low-infe",	deployableItemakinの更高を参照するための名前 を開印度は領域報 上記の-responTrypes-に定機した場合に使用可能な入力のインターフェース機関 上記の-responTrypes-に定機した場合に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して基金に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して上記の-responTratefraceTrypes-との処理は作品を含かpes-を使用する場合に、デ プロイに影響が構成の名前 上記の-responTrypes-に搭載して上記の-responTratefraceTrypes-との処理はfraceTrypesを使用する場合のファ	
{   "name": "Rem1",   "regionType": "ta",   "inputInterfaceType": "host100gether",   "outputInterfaceType": "host100gether",	deployable/terndxがの要素を参照するための名前 を無限性な経過期 上記の-creptonTypeンに配慮した場合に使用可能な人力のインターフェース専列 上記の-creptonTypeンに配慮した場合に使用可能な出力のインターフェース専列 上記の-creptonTypeンに促縮して場合と使用可能な出力のインターフェース専列 上記の-creptonTypeンに促縮して決定の-creptonTypeンと-couputInterfaceType>を使用する場合に、デ プロイに必要が参加を指	
{     "name": "flem1";     "regonType": "14";     "noputinterfaceType": "host100gether";     "outputinterfaceType": "host100gether",     "configName": "gpufunc-config-low-infer",     "specName": "spec1" }; {	deployable httms:が少更素を参照するための名前 を推測性な話場類 上記の-responType>に設備した場合に使用可能な入力のインターフェース専列 上記の-responType>に記憶した場合に使用可能な出力のインターフェース専列 上記の-responType>に記憶した場合に使用可能な出力のインターフェース専列 上記の-responType>に記憶しておき、ためで、中央の虹ボルギルでデッタンを使用する場合に、デ フロイに必要が扱わる形容 上記の-responType>に記憶して上記の、input interfaceType>と <output interfacetype="">を使用する場合のファンクションのスペック情報の名前</output>	
{     "name": "Rem1",     "regonType": "Net TLOGgether",     "noutInterfaceType": "host LOGgether",     "outputInterfaceType": "host LOGgether",     "configName": "gputInc-config-low-infer",     "spec Name": "spec L"     },     {         'name": "Rem1",     }	deployableItemakinの更高を参照するための名前 を開印度は領域報 上記の-responTrypes-に定機した場合に使用可能な入力のインターフェース機関 上記の-responTrypes-に定機した場合に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して基金に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して上記の-responTratefraceTrypes-との処理は作品を含かpes-を使用する場合に、デ プロイに影響が構成の名前 上記の-responTrypes-に搭載して上記の-responTratefraceTrypes-との処理はfraceTrypesを使用する場合のファ	
{     "name": "Rem1",     "negonType": "Net",     "negonType": "Net 100gether",     "noutInterfaceType": "host 100gether",     "corfigName": "goufunc-config-low-infer",     "spec*Name": "spec1"     };     {         "name": "Rem1",         "negonType": "Net",         "negonType": "n	deployableItemeds呼の更高を参照するための名前 位層可能と呼吸をは3種種別 上足の-crepporTypes-LC機能した場合に使用可能とはカカウインターフェース機別 上足の-crepporTypes-LC機能した場合に使用可能とはカカウインターフェース機別 上足の-crepporTypes-LC機能した場合に使用可能とはカウインターフェース機別 上足の-crepporTypes-LC機能した上足の-creputInterfaceTypes-と conquatInterfaceTypes-を使用する場合に、デ プロイに受視が開発の名前 上足の-crepporTypes-LC機能した上足の-creputInterfaceTypes-と conquatInterfaceTypes-を使用する場合のファ ンクションのスペック情報の名前 deployableItemeds/Hの要素を参照するための名前 を機能的なよびは複雑別 上足の-crepporTypes-LC機能した場合に使用可能と入力のインターフェース機関	
{     "name": "Rem1",     "regonType": "181",     "neutIntEndertype": "host100gether",     "outputInterfaceType": Thost100gether",     "corfgName": "gosfunc-config-low-infer",     "spec.Name": "spec.1"     "},     {         "name": "Rem1",         "repoinType": "181",     "regonType": "181",     "neutInterfaceType": "mem",     "outputInterfaceType": "host100gether",	deployabilitizendが例の要素を参照するための名前 企業可能では結婚的 上記の-creptompeesに発揮した場合に使用可能は入力のインターフェース機可 上記の-creptompeesに記憶した場合に使用可能は出力のインターフェース機可 上記の-creptompeesに記憶した場合に使用可能は出力のインターフェース機可 上記の-creptompeesに記憶した場合に使用可能は出力のインターフェース機可 上記の-creptompeesに記憶して上記の-creptutinerfaceTypeeを使用する場合に、デ ノフトインを受ける例の名前 上記の-creptompeesに記憶して上記の-creptutinerfaceTypeeを使用する場合のファ ンクションのスペップ開催の名前 上記の-creptompeesに記憶しておいた名称 を確認しまない。 上記の-creptompeesに記憶して発音に使用可能は入力のインターフェース機可 上記の-creptompeesに提修して表情に使用可能は入力のインターフェース機可	
{     "name": "Rem1",     "negonType": "Net",     "negonType": "Net 100gether",     "noutInterfaceType": "host 100gether",     "corfigName": "goufunc-config-low-infer",     "spec*Name": "spec1"     };     {         "name": "Rem1",         "negonType": "Net",         "negonType": "n	deployable/bramachinの資産を参照するための名前 配面可能は可能観知 上記の-respont/pse> に配慮した場合に使用可能な入力のインターフェース模別 上記の-respont/pse> に配慮した場合に使用可能なよ力のインターフェース模別 上記の-respont/pse> に配慮した場合に使用可能な出力のインターフェース模別 上記の-respont/pse> に配慮した場合に使用では配けったジャランエース模別 上記の-respont/pse> に配慮して上記のよりのはInterfaceStype> を使用する場合に、デ プロイに使せば極めた場別 上記の-respont/pse> にご確して上記のようのはInterfaceStype> を使用する場合のファ ンクシュンのスペック開催の名前 deployable/bramachinの要素を参照するための名前 を無限のはInterfaceStypeを表を使用する場合の方向 を無限のはInterfaceStype> に記慮した場合に使用で成立したクーフェース模別 上記の-respont/pse> にご確しした場合に使用可能な力かのインターフェース模別 上記の-respont/pse> にご確しして記念した場合に使用的な対力がインターフェース機関 上記の-respont/pse> にご確しして記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを受けることで表現して記念した。アフィースを使りませんからないました。アフィースを使りませんからないました。アフィースを使ります。アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする場合に、アフィースを使りまする。アフィースを使りまする場合に、アフィースを使りまする。アフィースを使りまする	
{     "name": "Rem1",     "regonType": "181",     "neutIntEndertype": "host100gether",     "outputInterfaceType": Thost100gether",     "corfgName": "gosfunc-config-low-infer",     "spec.Name": "spec.1"     "},     {         "name": "Rem1",         "repoinType": "181",     "regonType": "181",     "neutInterfaceType": "mem",     "outputInterfaceType": "host100gether",	deployableItemeds特が更高を参照するための名前 位置可定式は短線別 上足の-responTypes-LC機能した場合に使用可能なよ力のインターフェース機別 上足の-responTypes-LC機能した場合に使用可能なはカナのインターフェース機関 上足の-responTypes-LC機能した場合に使用可能なはカナのインターフェース機関 上足の-responTypes-LC機能した上記の-inquatinterfaceTypes-と-confputInterfaceTypes-を使用する場合に、デ プロイに必要が解める名前 上記の-responTypes-LC機能した上記の-inquatinterfaceTypes-と-confputInterfaceTypes-を使用する場合のファ ンクションのスペック情報の名前 deployableItemeds特の需要を参照するための名前 位置的対象が対象が特別をは対象が表現した場合に使用可能な入力のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能な入力のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能な认为のインターフェース機関 上記の-responTypes-LC機能した機能に使用可能な认为のインターフェース機関 上記の-responTypes-LC機能した機能に使用可能な认为のインターフェース機関 上記の-responTypes-LC機能した機能に使用可能な対象が分かってジャーフェース機能	
{	deployableItemeds呼の更高を参照するための名前 色層可能と呼吸をは関係的 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカナウインターフェース機関 上足の-responTypes-LC機能した上足の-inqualiterfaceTypes-と-conqualiterfaceTypes-を使用する場合のファンクションのスペック情報の名前 上型の-responTypes-LC機能した場合に使用可能と认为のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能と认为のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能とはカウィンターフェース機関 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用で起いませんからいった。 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用である。 上記の-responTypes-LC機能した影響という同様に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響という同様に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響という同様に使用可能と対力を対象を対力を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を	
{ "name": "Rem1", "regonType": "18", "publisher Exceptes": "host 100 gether", "output/interfaceType": "host 100 gether", "output/interfaceType": "host 100 gether", "configName": "gopfun-config-low-infer", "specName": "spec1"  ["name": "rem1", "regonType": "18", "reporType": "18", "reporType": "18", "specMarterfaceType": "host 100 gether", "configName": "gopfun-config-low-infer", "specName": "gopfun-config-low-infer", "specName": "spec1"  }	deployableItemeds呼の更高を参照するための名前 色層可能と呼吸をは関係的 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウインターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカナウインターフェース機関 上足の-responTypes-LC機能した上足の-inqualiterfaceTypes-と-conqualiterfaceTypes-を使用する場合のファンクションのスペック情報の名前 上型の-responTypes-LC機能した場合に使用可能と认为のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能と认为のインターフェース機関 上記の-responTypes-LC機能した場合に使用可能とはカウィンターフェース機関 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用で起いませんからいった。 上記の-responTypes-LC機能した影響に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響に使用である。 上記の-responTypes-LC機能した影響という同様に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響という同様に使用可能と対力のインターフェース機関 上記の-responTypes-LC機能した影響という同様に使用可能と対力を対象を対力を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を対象を	
{ "name": "Rem1", "regonType": "16", "publisher Exceptes" "host 100 gether", "bubutisher faceType": "host 100 gether", "outputisher faceType": "host 100 gether", "configName": "gopfun-config-low-infer", "specName": "spec1"  "name": "fem1", "regonType": "16", "regonType": "16", "specName": "gopfun-config-low-infer", "configName": "gopfun-config-low-infer", "specName": "gopfun-config-low-infer", "specName": "spec1"	deployableItemaを持つ変異を参照するための名前 を確認されば機器が 上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関 上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関 上記の-responTypes-に定機した場合に使用可能な対力のインターフェース機関 上記の-responTypes-に定機した場合に使用可能な対力のインターフェース機関 上記の-responTypes-に定機して上記の-reputInterfaceStypes-と-couputInterfaceStypesを使用する場合に、デ プロイにの影響が終める前 上記の-responTypes-に定機して上記の-reputInterfaceStypes-と-couputInterfaceStypesを使用する場合のファ ンクションのスペック情報の名前 上記の-responTypes-に定機した場合に参加可能な対力がインターフェース機関 上記の-responTypes-に定機した場合に参加可能な対力がインターフェース機関 上記の-responTypes-に定機したことがの-reputInterfaceStypes-と-couputInterfaceStypesを使用する場合に、デ フロイドの影響が影響が発	現 <i>在の</i> 試作では未使用
{ "rame": "Rem1", "regionType": "14", "nouthirefaceType": "host100gether", "outputterfaceType": "host100gether", "outputterfaceType": "host100gether", "configName": "gopfun-config-low-infer", "specName": "spec1"  "tame": "rem1", "regionType": "14", "inputturiefaceType": "host100gether", "outputterfaceType": "mem", "outputterfaceType": "host100gether", "configName": "gopfun-config-low-infer", "specName": "spec1"  } }	deployableIteme記憶の受責を参照するための名前  を開印をは対域機関  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能なよ力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  プロイに受ける場合を指揮の名前  上記の-responTypes-に定機して上記の-inpudInterfaceType>と<のutputInterfaceType>を使用する場合のファ ンクションのスペック情報の名前  位性のyableIteme記りの要素を参照するための名前  を確認が定式機構関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機してよるこの中間ではrefaceType>と使用する場合にデ ファンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアングミコンのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペックを表現しためまする。アングランのスペックを表現しためまする。アングランのスペックを表現しためまする。アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんで	現在の試作では未使用
{     "name": "Rem1",     "regonType": "Net 100gether",     "noutultrefraceType": "host 100gether",     "outputtrefraceType": "host 100gether",     "configName": "gpufun-config-low-infer",     "spec Name": "spec 1"     "},     {         "name": "Rem1",         "regonType": "Net,         "noutultrefraceType": "mem",     "outputtrefraceType": "host 100gether",     "configName": "gpufun-config-low-infer",     "spec Name": "spec 1"     "spec Name": "spec 1"     "spec Name": "spec 1"     "mande": "spec 1",     "mande": "spec 1"	deployableItemedsが少更高を参照するための名前 位置可定は領域間  上記の-responTypesとに簡単した場合に使用可能は入力のインターフェース推列 上記の-responTypesとに簡単した場合に使用可能は入力のインターフェース推列 上記の-responTypesとに簡単した場合に使用可能は対力のインターフェース推列 上記の-responTypesとに簡単した基色の cinquitinterfacelTypesと condputitinterfacelTypesを使用する場合に、デ プロイに最近が開始の名前 と記の-responTypesとに簡単した基色の cinquitinterfacelTypesと condputitinterfacelTypesを使用する場合のファ ンクションのスペック情報の名前 位置的では自然性が関連した場合に使用可能と入力のインターフェース推列 上記の-responTypesとに簡単した場合に使用可能と入力のインターフェース推列 上記の-responTypesとに簡単した場合に使用可能とは入力のインターフェース推列 上記の-responTypesとに指揮した上記の cinquitinterfacelTypesと condputitinterfacelTypesと使用する場合に、デ プロイに急性が開始の名前 上記の-responTypesに指揮した上記の cinquitinterfacelTypesと condputitinterfacelTypesと使用する場合にファンクションのスペック情報の名前 以下Bey valueがである。	現在の試作では未使用
{     "name": "Rem1",     "regonType": "14",     "regonType": "14",     "noutInterfaceType": "host100gether",     "outputInterfaceType": "host100gether",     "configName": "goutun-config-low-infer",     "specName": "spec1" },  {     "regonType": "M",     "regonType": "M",     "noutputInterfaceType": "host100gether",     "outputInterfaceType": "host100gether",     "specName": "spec1",     "specName": "spec1" }  spec: [     "rame": "spec1",	deployableIteme記憶の受責を参照するための名前  を開印をは対域機関  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能なよ力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  プロイに受ける場合を指揮の名前  上記の-responTypes-に定機して上記の-inpudInterfaceType>と<のutputInterfaceType>を使用する場合のファ ンクションのスペック情報の名前  位性のyableIteme記りの要素を参照するための名前  を確認が定式機構関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な出力のインターフェース機関  上記の-responTypes-に定機してよるこの中間ではrefaceType>と使用する場合にデ ファンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアンクミコンのスペック情報の名前  コアングミコンのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペック情報  コアングランのスペック情報  コアングランのスペックを表現した。日本では、アングランのスペック情報  コアングランのスペックを表現しためまする。アングランのスペックを表現しためまする。アングランのスペックを表現しためまする。アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんでは、アングランのスペックを表現しためませんで	現在が出る方式
{     "mame": "Rem1",     "regonType": "16",     "regonType": "16",     "poutInterfaceType": "host100gether",     "outputInterfaceType": "host100gether",     "configName": "goutunc-config-low-infer",     "specName": "goutunc-config-low-infer",     "specName": "spect"     ""specName": "Nem1",     "regonType": "Nem",     "subditterfaceType": "host100gether",     "sopcMame": "spect ""     "specName": "spect"     "spect";     "manCapeType": "spect",     "manCapeTypeType Spect",     "manCapeTypeTypeType Spect",     "manCapeTypeTypeTypeTypeTypeTypeTypeTypeTypeTy	deployabeltemedを呼び要素を参考するための名前  を確認をは関係数  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力力のインターフェース機関  上記の-responTypes-に定機して上記の-responTypes-に変した機関では、デ  上記の-responTypes-に定機して上記の-responTypes-に変した場合のファ  ンクションのスペック情報の名前  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機して上記の-responTypes-とで機関して影響を使用する場合にデ  上記の-responTypes-に定機してよるのののはmereforExpes-とのdoubniterforExpes-を使用する場合のファ  フランコンのスペック情報の名前  コアンクコンのスペック情報の名前  コアンクコンのスペック情報の名前  コアンクコンのスペック情報の名前  明するリソースの配子機  使用するリソースの配子機  電用するリソースの配子機  電用するリソースの配子機  基本の最大性解析を行か(9)	現在の試作では未使用
{     "name": "Rem1",     "negonType": "Net 100gether",     "nouturiter face Type": "host 100gether",     "output Inter face Type": "host 100gether",     "configName": "gputure-config-low-infer",     "spec Name": "spec 1" }, {     "name": "Rem1",     "negonType": "Net",     "noutput Inter face Type": "host 100gether",     "configName": "gputure-config-low-infer",     "spec Name": "spec 1"     "spec Name": "spec 1"     "spec Name": "spec 1"     "ype Name": "spec 1"     "ype Name": "spec 1"     "name": "spec 1",     "name": "spec 1",     "name": "spec 1",     "name": "spec 1",     "name Oats Thouse 1.1,     "name Oats Thouse 2.1,     "n	deployable/timeds/Prop事業を参照するための名前 を確可能は近端機関 上記の-region/poeとに関連した場合に使用可能な入力のインターフェース機関 上記の-region/poeとに関連した場合に使用可能な入力のインターフェース機関 上記の-region/poeとに関連した場合に使用可能な出力のインターフェース機関 上記の-region/poeとに関連して上記のに関連した場合に使用では対したインターフェース機関 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合に、デ プロイに必要が締めた前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合のファ ンクションのスペック開始の名前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合のファ フクションのスペック開始の名前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) e-を使用する場合に、デ フラントのでは、アルスのよりを表して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合に、デ フランコンのスペルの分解 以下intervalueが与などののオブジェクトを要素とする影響を文字列にした機 可用するリンースの最大機 構工のサンースの最大機 基本の最大的の最小機 基本の最大的関係の機関を表が関係に対している。 第本の最大的関係を表が関係と対している。 第本の最大的表 第本の最大的表 第本の最大的表 第本の表大的表	現在心試作では未使用
{     "rame": "Rem1",     "regonType": "161",     "regonType": "161",     "putInterEnceType: "host100gether",     "outputInterEnceType: "host100gether",     "configName": "goufunc-config-low-infer",     "specName": "spec1" },  "spec Name": "spec1" ,     "spec1": "16",     "spec1	deployabeltemedを呼び要素を参考するための名前  を確認をは関係数  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な入力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力力のインターフェース機関  上記の-responTypes-に定機して上記の-responTypes-に変した機関では、デ  上記の-responTypes-に定機して上記の-responTypes-に変した場合のファ  ンクションのスペック情報の名前  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機した場合に使用可能な力のインターフェース機関  上記の-responTypes-に定機して上記の-responTypes-とで機関して影響を使用する場合にデ  上記の-responTypes-に定機してよるのののはmereforExpes-とのdoubniterforExpes-を使用する場合のファ  フランコンのスペック情報の名前  コアンクコンのスペック情報の名前  コアンクコンのスペック情報の名前  コアンクコンのスペック情報の名前  明するリソースの配子機  使用するリソースの配子機  電用するリソースの配子機  電用するリソースの配子機  基本の最大性解析を行か(9)	現在の試作では未使用
{     "name": "item1",     "negion"/pet": 144",     "negion"/pet": 144",     "negion*/pet": 144",     "negion*/pet": 144",     "spectivitierface(Typet": host100gether",     "configName": "sputunc-config-low-inder",     "specName": "spec1" },     """     "name": "spec1"     "name": "spec1"     "negion*/pet": "mem",     "output/itemface*/pet": "mem",     "output/itemface*/pet": "host100gether",     "configName": "spec1"     "specName": "spec1" }    "specName": "spec1" }  spec: "{     "name": "spec1",     "name	deployable/timeds/Prop事業を参照するための名前 を確可能は近端機関 上記の-region/poeとに関連した場合に使用可能な入力のインターフェース機関 上記の-region/poeとに関連した場合に使用可能な入力のインターフェース機関 上記の-region/poeとに関連した場合に使用可能な出力のインターフェース機関 上記の-region/poeとに関連して上記のに関連した場合に使用では対したインターフェース機関 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合に、デ プロイに必要が締めた前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合のファ ンクションのスペック開始の名前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合のファ フクションのスペック開始の名前 上記の-region/poeとに関連して上記の-reputInterface(Poe) e-couptInterface(Poe) e-を使用する場合に、デ フラントのでは、アルスのよりを表して上記の-reputInterface(Poe) e-couptInterface(Poe) を使用する場合に、デ フランコンのスペルの分解 以下intervalueが与などののオブジェクトを要素とする影響を文字列にした機 可用するリンースの最大機 構工のサンースの最大機 基本の最大的の最小機 基本の最大的関係の機関を表が関係に対している。 第本の最大的関係を表が関係と対している。 第本の最大的表 第本の最大的表 第本の最大的表 第本の表大的表	現在の試作では未使用

「2.YAML説明」のDataFlowのYAMLファイルの設定内容のうち、各処理モジュールのPodが使用するネットワーク情報(IPアドレス・ポート番号) の設定について説明する

以下のAに全処理モジュール共通の設定を、以下のBに各処理モジュール毎の設定内容を記載する

DataFlow YAML	説明
apiVersion: example.com/v1	
kind: DataFlow	
metadata:	
name: "df-test-3-1-1-1"	ユーザが任意に設定する
namespace: "test01"	ユーザが任意に設定する
spec:	
functionChainRef:	
name: "cpu-decode-cpu-filter-resize-2types-high-infer-chain"	DataFlowが利用するFunctionChainのmetadata.Name
namespace: "chain-imgproc"	DataFlowが利用するFunctionChainのmetadata.Namespace
requirements:	スケジューリング時に満たす必要がある要件を記載する
all:	ファンクションチェイン全体の要件を記載
capacity: 15	想定負荷量(fps)を記載
functionUserParameter:	
- functionKey: decode-main	CPUデコードFunctionの識別子
userParams:	
ipAddress: 192.174.90.101/24	自身のIPアドレス。 Podの2nd NICのIPアドレスとして設定される
inputPort: 5004	自身のポート番号
outputIPAddress: 192.174.90.111	送信先 (CPUフィルタリサイズ) のIPアドレス
outputPort: 15000	送信先(CPUフィルタリサイズ)のボート番号
- functionKey: filter-resize-high-infer-main	CPUフイルタリサイズFunctionの識別子
userParams:	
ipAddress: 192.174.90.111/24	自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
inputPort: 15000	自身のポート番号
outputIPAddress: 192.174.90.121	送信先 (コピー分岐) のIPアドレス
outputPort: 16000	送信先 (□ピー分岐) のポート番号
- functionKey: copy-branch-main	コピー分岐Functionの識別子
userParams:	
ipAddress: 192.174.90.121/24	自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
inputIPAddress: 192.174.90.121	自身のIPアドレス。前段のファンクションとのTCP接続の確立のために必要
inputPort: 16000	自身のボート番号
branchOutputIPAddress: 192.174.90.141,192.174.90.142	送信先 (GPU高度推論1、GPU高度推論2) のIPアドレスをカンマ区切りで指定
branchOutputPort: 17000,18000	送信先(GPU高度推論1、GPU高度推論2)のポート番号をカンマ区切りで指定
- functionKey: infer-1	GPU高度推論Function1の識別子
userParams:	
ipAddress: 192.174.90.141/24	自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
inputIPAddress: 192.174.90.141	自身のIPアドレス。GStreamerのビデオ処理のコマンド実行(fpgdepay)に必要
inputPort: 17000	自身のポート番号
outputIPAddress: 192.174.90.10	送信先(映像受信ツール)のIPアドレス
outputPort: 2001	送信先(映像受信ツール)のボート番号
- functionKey: infer-2	GPU高度推論Function2の識別子
userParams:	
ipAddress: 192.174.90.142/24	自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
inputIPAddress: 192.174.90.142	自身のIPアドレス。GStreamerのビデオ処理のコマンド実行(fpgdepay)に必要
inputPort: 18000	自身のボート番号
outputIPAddress: 192.174.90.10	送信先(映像受信ツール)のIPアドレス
outputPort: 2002	送信先(映像受信ツール)のポート番号
userRequirement; user-requirement	DataFlowのスケジューリング用の各種設定情報の取得のために参照するUserRequirementのConfigMapの
accordance and reduce 1616	metadata.nameを指定

# A.全処理モジュール共通の設定内容について

- ・自身がPod上で動作する処理モジュールであり、
- 自身に対する送信元が別のPod上で動作する処理モジュールまたは映像配信ツールの場合は、ipAddressとinputPortを設定する
- ・自身がPod上で動作する処理モジュールであり、
- 自身からの送信先が別のPodで動作する処理モジュールまたは映像受信ツールの場合は、ipAddressとoutputIPAddressとoutputPortを設定する
- ・一部の処理モジュールでは、当該処理モジュールの固有の処理のために、inputIPAddressを設定する
- ・ipAddress、inputIPAddress、outputIPAddressを設定する場合、
- 「OpenKasugai-Controller-InstallManual」の「8.9 SR-IOVのVF作成および管理」で VFを作成した100GNICの物理IPアドレスと共通のサブネットのIPアドレスを設定する
- ・Pod上で動作しないFPGAデコード/FPGAフィルタリサイズの処理モジュールの場合は、上記いずれの設定も不要

# B.各処理モジュール毎の設定内容について

A.CPUデコードを使用する場合

functionKey: decode-main userParams:

① ipAddress: 192.174.90.101/24

自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される ② inputPort: 5004

③ outputIPAddress: 192.174.90.111 ④ outputPort: 15000

自身のポート番号

送信先のIPアドレス 送信先のIPアドレス

②についての補足:ボート番号の値は映像配信ツールで指定したCPUデコードアプリのボート番号を記載する。

以下はOpenKasugai-Demoの1.3.3節(1)の①を抜粋

./start\_gst\_sender.sh /opt/video/pocdemo\_movie/day\_scene/d1\_12\_Videvo-4\_2K\_160929\_057\_London\_WestminsterBridge7\_1080p\_5min\_conv\_4K\_8Mbps\_15fps.mp4 <mark>192.174.90.101 5004</mark> 1 \${sleep\_time:-3}↔

./start\_gst\_sender.sh /opt/video/pocdemo\_movie/day\_scene/d2\_06\_Pexels-15\_4K\_pexels-creativ-medium-5607960\_5min\_conv\_4K\_6Mbps\_15fps.mp4

192.174.90.102 5004 1 \${sleep\_time:-3}↔

# B.CPUフィルタリサイズを使用する場合

functionKey: filter-resize-xxx-infer-main 高度推論用の場合は filter-resize-high-infer-mainを、軽量推論用の場合は filter-resize-low-infer-mainをfunctionKeyとして指定 userParams:

① ipAddress: 192.174.90.111/24 自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される

自身のポート番号 ② inputPort: 15000 ③ outputIPAddress: 192.174.90.121 送信先のIPアドレス ④ outputPort: 16000 送信先のボート番号

C.コピー分岐を使用する場合

```
functionKey: copy-branch-main
         userParams:
       ① ipAddress: 192.174.90.121/24
                                                                 自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
                                                                 自身のIPアドレス。前段の処理モジュールとのTCP接続の確立のために利用(サブネットマスクの設定は不要)
       ② inputIPAddress: 192.174.90.121
       ③ inputPort: 16000
                                                                 自身のポート番号

    branchOutputIPAddress: 192.174.90.141,192.174.90.142

                                                                 送信先のIPアドレスをカンマ区切りで指定
       ⑤ branchOutputPort: 17000,18000
                                                                 送信先のボート番号をカンマ区切りで指定
         ②についての補足:①と同じIPアドレスを設定する(サブネットマスクの記載は不要)
         ④についての補足:送信先の分岐数分、IPアドレスをカンマ区切りで設定する
         ⑤についての補足:送信先の分岐数分、ボート番号をカンマ区切りで設定する
D.GPU推論を使用する場合
         functionKey: xxx-infer-main または infer-[n]
                                                        DataFlowにコピー分岐なしの場合、高度推論の場合はhigh-infer-mainを、軽量推論の場合はlow-infer-mainをfunctionKeyとして指定
         userParams:
                                                        DataFlowにコピー分岐ありの場合、分岐した内の何個目のGPU推論であるかに応じて、infer1またはinfer2をfunctionKeyとして指定
       ① ipAddress: 192.174.90.141/24
                                                        自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される
                                                        自身のIPアドレス。GStreamerのビデオ処理のコマンド実行(fpgdepay)に利用(サブネットマスクの設定は不要)
       ② inputIPAddress: 192.174.90.141
                                                       自身のポート番号
       3 inputPort: 16000
       4 outputIPAddress: 192.174.90.10
                                                       映像受信ツールのIPアドレス
       ⑤ outputPort: 2001
                                                       映像受信ツールのボート番号
         ②についての補足:①と同じIPアドレスを設定する(サブネットマスクの記載は不要)
         ④についての補足: IPアドレスの値は映像受信ツールが使用するK8s control planeの100GNICのIPアドレスを指定する
                          K8s control planeで以下のコマンドを実行し、IPアドレスを確認する。
          1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 adisc noqueue state UNKNOWN group default alen 1000
            link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00
            inet 127.0.0.1/8 scope host lo
              valid_lft forever preferred_lft forever
            inet6::1/128 scope host
              valid Ift forever preferred Ift forever
          2: ens1f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
            link/ether b4:96:91:9d:79:80 brd ff:ff:ff:ff:ff
            inet 10.38.119.15/24 brd 10.38.119.255 scope global ens1f0
              valid Ift forever preferred Ift forever
            inet6 fe80::b696:91ff:fe9d:7980/64 scope link
              valid_lft forever preferred_lft forever
          ~省略~
          7: ens3f1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP group default qlen 1000
            link/ether 0c:42:a1:6d:65:35 brd ff:ff:ff:ff:ff
            inet 192.174.90.10/24 brd 192.174.91.255 scope global ens3f1
              valid Ift forever preferred Ift forever
            inet6 fe80::e42:a1ff:fe6d:6535/64 scope link
              valid_lft forever preferred_lft forever
         ⑤についての補足:ボート番号の値は評価手順の映像受信開始手順で指定したボート番号を記載する。
           sample-demoの例: 2001か2002を指定する。
         以下はOpenKasugai-Demoの1.3.2節の(1)の①を抜粋
            (1) 映像受信の起動方法 ←

① 映像受信ツールの配備と受信用スクリプト(高度推論結果受信×2)の作成↓
                    (初回のみ実施、2回目以降は手順①は不要)\downarrow
                    $ cd ~/openkasugai-controller/sample-functions/utils/rcv_vide_tool/
                   $ kubectl create ns test
                   $ kubecti treater is test"

$ kubectl apply -f rcv_video_tool.yaml#

$ kubectl exec -n test -it rcv-video-tool-XXX -- bash ※pod名は環境に合わせて

変更#
                   # vi start_test1.sh ※以下を貼り付けて保存する↓
#!/bin/bash -x ↓
                    for i in `seq -w<mark>01 02`</mark>↔
                      gst-launch-1.0 -e udpsrc buffer-size=21299100 mtu=8900 port=20${i} !
                   gst-launch-1.0 -e udpsrc buffer-size=21299100 mtu=9900 [port=2085]] !
'application/x-rtp, media-(string)/video, clock-rate(-int)90800, encoding-
name-(string)RAW, sampling-(string)BGR, depth-(string)8, width-(string)1280,
height-(string)1280, payload=(int)96' ! rtpvrawdepay ! queue ! videocvnevrt !
'video'x-raw, format-(string)1420' ! openh264en ! 'video'x-rab264 stream-
format-byte-stream, profile-(string)high' ! perf name-stream5[i] ! h264parse !
                    qtmux
                           ! filesink location=/tmp/output_st${i}.mp4
                                                                             sync=false
                    /tmp/rcv_video_tool_st${i}.log & ↔
                   done +
                   # chmod +x start_test1.sh ↔
# exit↔
E.Glueを使用する場合 ※2.YAML説明のDataFlowでは利用していない処理モジュール
```

# F.FPGAデコード/FPGAフィルタリサイズを使用する場合

functionKey: glue-fdma-to-tcp-main

② glueOutputIPAddress: 192.174.90.141

設定不要

userParams:
① ipAddress: 192.174.90.131/24

3 glueOutputPort: 16000

送信先のIPアドレス

送信先のボート番号

自身のIPアドレス。Podの2nd NICのIPアドレスとして設定される

# B. 提供した資材から編集が必要なファイル

決め打ち領域情報 ※kBsクラスタ内で使用する全領域に関して、各領域の領域種別を記載する必要あり。

predetermined-region.json	説明	環境に合わせた	
• •		変更が必要※	
			領域単位で設定する
{	1領域分の情報		
"nodeName": "worker0",	その領域があるノードのノード名	0	
"deviceUUID": "21330621T013",	その領域があるデバイスの施別情報	0	<ul> <li>-FPGAの場合: "Is /dev/*"コマンドの結果を利用。FPGAdデバイスは"xpcie_\${FPGA·ID}"と表示されるので、\${FPGA·ID}の値を記載する</li> <li>-GPUの場合: "nvidia-smi -L"コマンドの結果を利用。デバイス毎にUUID(例: GPU-b8b4f1f5-bf51-eaa3-6ec4-97190b7f6c98)が出力されるのでその値を記載する</li> </ul>
·			・CPUの場合: "0"を記載する(現状各サーバで仮想的に1つとみなしているため、固定値で良い)
"subDeviceSpecRef": "0",	その領域の領域名	0	- FPGAの場合: "\${lane指号}"を記載する - CPU/GPUの場合: deviceTypeと同じ値を記載する
"regionType": "alveou250-0100001c-2lanes-0nics"	その領域の領域種別	0	- FPCAの場合: 以下のフォーマかで記載: 「**5「デイス(福別)* + *** + **(指b-sd)* + *** + **5 (lane数)* * * *anes* + *** + **5 (NIC数)* + *nics*] サンプルとに「理じているFPCA回路(Pochessugai-Controller-Installantanicon) の称に定動と用いる場合は、 デバス機別 **alveou250*, 能b-sid=**010001c*, lane数**2*, NIC数**0**で固定点ので、 regionTypeは必ずalveou250・100001c*, lanes-Onics*となる。 - **CPU/GPUの場合: deviceToxe(可能)を必定者を
},			
{	1領域分の情報		
"nodeName": "worker0",		0	
"deviceUUID": "21330621T01J",		0	
"subDeviceSpecRef": "1",		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
},			
{	1領域分の情報		
"nodeName": "worker1",		0	
"deviceUUID": "21330621T00Y".		0	
"subDeviceSpecRef": "0",		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
),			
{	1領域分の情報		
"nodeName": "worker1".	-97-972-1118	0	
"deviceUUID": "21330621T00Y",		0	
"subDeviceSpecRef": "1".		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
\			
1	1領域分の情報		
"nodeName": "worker1",	A SPECIAL AND STATE	0	
"deviceUUID": "qpu-702fb653-43a4-732d-6bc4-7b3487696c90"		0	
"subDeviceSpecRef": "a100",		0	
"regionType": "a100"		0	GPU/CPUの領域種別の値はデバイス種別(deviceType)と同等
1		0	G. G. G. SP. SP. SP. SP. SP. SP. SP. SP. SP. SP
1			
	II.		

# A. 全てのDFで共通的に使用可能なファイル(環境に合わせた変更が不要でそのまま使える)

デバイスタイプマッピング情報 自動取得したデバイスのモデル名からDeviceTypeに変換するためのマッピング情報。

※サンプルデータには以下の6種類のデバイス(Alveo U250, NVIDIA GPU T4, NVIDIA GPU A100, Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz, Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz, Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz)について記載している。それ以外のデバイスを使用する場合は追記が必要。

している。てれ以アのアハイスを使用する場合は足記が必要。		
	D0-73	備考
K	1デバイス分の情報	
"inputDeviceType": "ALVEO U250 PQ",	インフラから取得出来るデバイス情報	「自動収集&CM作成ツール用事前情報チェックツール(DeviceInfoCheck.sh)を実行して得られた値を設定している
"outputDeviceType": "alveou250"	上記デバイスに対応するDeviceType	"Alveo系"は"alveo"決め打ち
},{	1デバイス分の情報	
"inputDeviceType": "Tesla T4",		「自動収集&CM作成ツール用事前情報チェックツール(DeviceInfoCheck.sh)を実行して得られた値を設定している
"outputDeviceType": "t4"		"Tesla T4"は"t4"決め打ち
}.{	1デバイス分の情報	
"inputDeviceType": "NVIDIA A100 80GB PCIe",		「自動収集&CM作成ツール用事前情報チェックツール(DeviceInfoCheck.sh)を実行して得られた値を設定している
"outputDeviceType": "a100"		"NVIDIA A100 80GB PCIe"は"a100"決め打ち
}.{	1デバイス分の情報	
"inputDeviceType": "Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz",		grep model.name /proc/cpuinfo   sort -u の結果を確認し、設定
"outputDeviceType": "cpu"		"Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz"は"cpu"決め打ち
}.{	1デバイス分の情報	
"inputDeviceType": "Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz",		grep model.name /proc/cpuinfo   sort -u の結果を確認し、設定
"outputDeviceType": "cpu"		"Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz"は"cpu"決め打ち
},{	1デバイス分の情報	
"inputDeviceType": "Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz",		grep model.name /proc/cpuinfo   sort -u の結果を確認し、設定
"outputDeviceType": "cpu"		"Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz"は"cpu"決め打ち
}]		

領域固有情報 領域毎の性能(配備容量と最大処理能力)を記載する。

- ・領域の性能は、FPGAならばデバイス種別(デバイスのモデル)と構込む子bitstreamによって変わる想定のため使用するデバイス種別×子bitstream分の記載が必要。CPU/GPUの領域は現状はデバイス種別のみによって変わる想定のため使用するデバイス種別分の記載が必要。サンプルデータには以下の6種類の領域について記載している。それ以外の領域を用いる場合は追記が必要。
   FPGAについては1種類の領域について記載。具体的には、Alveo U250にフィルタリサイズ用のbitstreamが書込まれている領域について記載

- ・GPUについては2種類の領域について記載。具体的には、2種類のデバイス(NYIDIA GPU T4, NVIDIA GPU A100)用の領域について記載
  ・CPUについては3種類の領域について記載。具体的には3種類のデバイス(Intel(R) Xeon(R) Gold 6346 CPU @ 2.60GHz と Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz)用の領域について記載

region-unique-info.json	説明	備考
{	1デバイス分の領域情報	Alveo U250にフィルタリサイズ用子bitstreamが書込まれている領域分
"subDeviceSpecRef": "0100001c",	デバイスに書込むオブジェクトのId	FPGAではそのデバイスに書いる子がtitstreamのId(ここではフィルタリウス用bitstreamのId)を記載。
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	laneOff
"regionName": "lane0",	当該領域の領域名(デバイス内で一意)	サンブル子bitstream(フィルタリサイズ用子bitstream)を使用する場合は"lane0"もしくは"lane1"決め打ち
"regionType": "alveou250-0100001c-2lanes-0nics",	当該領域の領域種別	FPGAの領域種別の値は以下のフォーマットで記載:「"\${デバイス種別}" + "-" + "\${朝bs-id}" + "-" + "\${lane数}" + "lanes" + "-" + "\${NIC数}" + "nics"]
"maxFunctions": 1,	当該領域でのFunction(回路/Pod)を配備容量	当該領域に搭載可能な最大Function(回路/Pod)数
"maxCapacity": 40	当該領域の最大処理能力	単位はfps. 値は暫定値。
},{		lane1分
"regionName": "lane1",		
"regionType": "alveou250-0100001c-2lanes-0nics",		
"maxFunctions": 1,		
"maxCapacity": 40		
)]		
.{	1デバイス分の領域情報	NVIDIA GPU T4用の領域分
"subDeviceSpecRef": "Tesla T4",	デバイスに書込むオブジェクトのId	GPUとCPUではデバイス種別(デバイスのモデル)名を記載
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	GPUはデバイス毎に1領域(デバイス全体を1つの領域)
"regionName": "t4",		"Tesla T4"は"t4"決め打ち
"regionType": "t4",		CPUとGPUではデバイス種別(deviceType)と同じ値
		このという。このでは、アインでは
"maxFunctions": 110,		
"maxCapacity": 40		単位はfps。 値は新定値。
}]		
v4.	1デバイス分の領域情報	NVIDIA GPU A100用の領域分
"subDeviceSpecRef": "NVIDIA A100 80GB PCIe",	デバイスに書込むオブジェクトのId	GPUとCPUではデバイス種別(デバイスのモデル)名を記載
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	GPUはデバイス毎に1領域(デバイス全体を1つの領域)
"regionName": "a100",		"NVIDIA A100 80GB PCIe"は"a100"決め打ち
"regionType": "a100",		CPUとGPUではデバイス種別(deviceType)と同じ値
"maxFunctions": 110,		当該領域に搭載可能な最大Function(回路/Pod)数。 偏は新定值。
		- Monte Control - Microsoft Control (Control - Control
"maxCapacity": 120		中国(4)内5- 测点测失测。
}]	Control (or () - ATTINUESD	THE PARTY OF THE P
K	1デバイス分の領域情報	Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz用の領域分
"subDeviceSpecRef": "Intel(R) Xeon(R) Gold 6346 CPU @ 3.10GHz",	デバイスに書込むオブジェクトのId	GPUとCPUではデバイス種別(デバイスのモデル)名を記載
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	CPUはデバイス毎に1領域(デバイス全体を1つの領域)
"regionName": "cpu",		CPUは"cpu"決め打ち
"regionType": "cpu",		CPUとGPUではデバイス種別(deviceType)と同じ値
"maxFunctions": 110,		当該領域C搭載可能な最大Function(回路/Pod)数。 値は暫定値。
"maxCapacity": 120		単位はfps. 値は新定値。
maxcapacity : 120		中山の中の世の日に世の
<i>t</i>	1デバイス分の領域情報	Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz用の領域分
"subDeviceSpecRef": "Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz",		Intel(K) Golfe 6.348 LPU 回 (5.00G)27月90日   GPUCPUではデバイス勝列デバイスのモデル)名を記載
	デバイスに書込むオブジェクトのId	
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	CPUはデバイス毎に1領域(デバイス全体を1つの領域)
"regionName": "cpu",		CPUは"cpu"決め打ち
"regionType": "cpu",		CPUとGPUではデバイス種別(deviceType)と同じ値
"maxFunctions": 110,		当該領域C搭載可能な最大Function(回路/Pod)数。値は無定値。
"maxCapacity": 120		単位はfos. 値は新定値。
\]		
fi .f	1デバイス分の領域情報	Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz用の領域分
"subDeviceSpecRef": "Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz",	エデバイスがい領域に再報 デバイスに書込むオブジェクトのId	Intel(k) Aeo(i(k) Goli 0530 ピヤウ 2.00512/myine(J)   GPUとつけじはデバイス権別(デバイスのモデル)名を記載
"functionTargets":[{	このオブジェクトを書込んだ場合に作られる領域の数だけ記載する	CPUはデバイス毎に1領域でデバイス全体を1つの領域)
"regionName": "cpu",	CV/1フフェントで自たがいた物口にTFO(IのII)機の分類にい記載する	CPUI*cpu'\\$\text{spt15}
regionType": "cpu",		CPUとGPUではデバイス権例(deviceType)と同じ値
"maxFunctions": 110,		当該領域に搭載可能な最大Function(回路/Pod)效。値は暫定値。
		= Table (Part Speed Sp
"maxCanacity": 120		
"maxCapacity": 120		中国は中の一般の部と知っ

### FunctionType特定用対応情報

使用するDeviceKindの数分のエントンを記載する
※サンプルデータには4種類のデバイスタイプ(alveou250, t4, a100, cpu)について記載している。それ以外のデバイスタイプを使用する場合は、デバイスタイプマッピング情報に追加デバイスタイプ分の情報を追記するとともに、本情報に追加デバイスタイプ分の情報の追記が必要。

追加ナバイスがの領域に関する情報の追記が必要。追記を行う場合の追記。		
functionkindmap.json		偏考
K		FPGAFunction用の情報
"deviceType": "alveou250",	key: WBFuncでのDeviceTypeの文字列	該当するDeviceType
"functionCRKind": "FPGAFunction"		deviceType="alveou250"ならFPGAなので"FPGAFunction"決め打ち
}.{	1種類のFunction系CR分の情報	GPUFunction用の情報
"deviceType": "t4",		該当するDeviceType(NVIDIA GPU T4向け)。
"functionCRKind":"GPUFunction"		deviceType="t4"ならGPUなので"GPUFunction"決め打ち
Ж	THE ARTON CONCOUNTS TO THE PARTY OF THE PART	GPUFunction用の情報
"deviceType": "a100",		該当するDeviceType(NVIDIA GPU A100向け)。
"functionCRKind":"GPUFunction"		deviceType="a100"ならGPUなので"GPUFunction"決め打ち
}.{	1種類のFunction系CR分の情報	CPUFunction用の情報
"deviceType": "cpu",		該当するDeviceType(各種cpu向け)。
"functionCRKind":"CPUFunction"		deviceType="cpu"ならCPUなので"CPUFunction"決め打ち
}]		

### ConnectionType特定用対応情報

使用するコネクションのType数分のエントリを記載する
※サンプルデータには2種類の接続種別(共有メモリ経由のPCIe接続とEthernet接続)について記載している。それ以外の接続種別を使う場合は追記が必要。

connectionkindmap.json	説明	備考
K	1種類のConnection系CR分の情報	PCIeConnection用の情報
"connectionMethod": "host-mem",	key: WBConnectionでのconnectionMethodの文字列	共有メモリ経由のPCIe接続を行うConnectionMethod
"connectionCRKind": "PCIeConnection"	value: Connection系CRの種類	connectionMethod="host-mem"ならPCIe接続なので"PCIeConnetion"決め打ち
M	1種類のConnection系CR分の情報	EthernetConnection用の情報
"connectionMethod": "host-100gether",		Ethernet接続を行うConnectionMethod
"connectionCRKind": "EthernetConnection"		connectionMethod="host-100gether"ならEther接続なので"EthernetConnetion"決め打ち
}]		

# Function固有情報 - 共通属性

\*\*\*FPOAのプァンクション種別毎にエントリを記載する。 \*\*\*GPUのファンクションについては現状記載不要(将来的に記載する方向に変更の可能性はあり)。
\*\*\*サンプルデータには"filter-resize"についてのみ記載している。それ以外のFPGAのファンクションを追加する場合は追記が必要。また、既存のファンクションについても性能値が変化した場合は更新が必要。

function-unique-info.json	説明	備考
[{	1Function分の情報	filter/resizeのFuntion用の情報
"functionID" : 0,	当該Function(回路/コンテナイメージ)の識別子	現状使っていないので値は空でも良い。
"functionName" : "filter-resize",	当該Function名	
"maxDataFlows": 8,	当該Functionに配備可能な最大DF(WBFunc)数	FPGA回路の場合は同時に提供可能なFunctionChannelID数を記載。
"maxCapacity": 40	当該Functionの最大処理能力	
N		

### Function固有情報 - filter/resize専用属性

\*\*\*FunctionChannelIDListは、filter/resize(F/R)用のピットストリームで書込んだ全filter/resizeファンクション(1FPGAデバイス毎に2ファンクション分)で提供するFunctionChannelIDの数だけエントリを記載する。
\*\*\*「RX'と「アント"と関いて、送信元も人は送信光との終終タイプが複数考えられる場合はそれぞれ用の値の定義が必要。

・RXで、IXに関し、ス造に方もしくは这倍がもの接続ゲイノが複数考えがも多高はそれぞれ用の側の定義が必要。 Her-resize-childbs.ison			
filter-resize-childbs.json	説明	· 하면 :	
{ 	filter/resize用bitstreamの1lane分のリソース情報	Ch. A. S. Waller Co. C.	
"functionKernels":[{	1laneに配備されるFunction1つ分で用意するリソースブールの情報	filter/resize用Bitstreamの場合はIlaneにつき1Functionのみ filter/resizeの場合Ilaneにつき1Functionなで、この簡はIneのId(のか1)になる。	
"partitionName": "0",	当該FunctionのId		
"functionChannelIDList": [0,1,2,3,4,5,6,7],	当該Functionで提供するFunctionChannelIDのリスト	当該Functionで提供するFunctionChannelIDを全て記載する。filter/resizeの場合は[0,1,2,3,4,5,6,7]固定で良い	
"functionChannelIDs":[{	1FunctionChannelID分のリソース情報	各FPGAFunctionに割り振られる各FunctionChannelID用に用意されたリソース情報。	
"functionChannelID": 0,	対象となるFunctionChannelID	上記functionChannelIDList内の値になる。	
"rx":{	当該FunctionChannelID向けに提供する受信用リソース群		
"protocol":{	通信可能なプロトコル毎に記載する		
"TCP":{	送信元との通信のプロトコルがTCPの場合に割り振るリソース群		
"port": 12300	送信元との通信で使用するボート番号		
).			
"DMA":{	送信元との通信のプロトコルがDMAの場合に割り振るリソース群		
"port": 12300.	送信元との通信で使用するボート番号		
"IldmaConnectorID": 0.	送信元とDMA通信する際に使用するLLDMAのコネクタId		
"dmaChannelID": 0	送信元とDMA通信する際に使用するDMAのチャネルId		
}			
}			
},			
"tx":{	上記FunctionChannelID向けに提供する送信用リソース群		
"protocol":{	通信可能なプロトコル毎に記載する		
"TCP":{	宛先との通信のプロトコルがTCPの場合に割り振るリソース群		
"port": 12300	宛先との通信で使用するボート番号		
},			
"DMA":{	宛先との通信のプロトコルがDMAの場合に割り振るリソース群		
"port": 12300,	宛先との通信で使用するボート番号		
"lldmaConnectorID": 0,	宛先とDMA通信する際に使用するLLDMAのコネクタId		
"dmaChannelID": 0	宛先とDMA通信する際に使用するDMAのチャネルId		
}			
,}			
}	1FunctionChannelID分のリソース情報		
"functionChannelID": 1,	1FunctionChannelID分のリソース情報 対象となるFunctionChannelID		
"rx":{	対象となるFunctionChannelID向けに提供する受信用リソース群		
1X 33	当該FunctionChannettD内がに提供する支信用サクースは		
,			
"tx":{	当該FunctionChanneIID向けに提供する送信用リソース群		
	コaxi directorie ineutolyty/に提供する広信用リケー人件		
1			
}.			
31			
M	1laneに配備されるFunction1つ分で用意するリソースブールの情報	filter/resize用bitstreamの場合は1laneにつき1Functionのみ	

# 3.CM作成に使用する入力データ(JSON)の説明

"partitionName": "1",	
}.	
1	
3	

フィルタリサイズファンクション名特定用対応情報
※FPGAでフィルタリサイズが高度推論向けか軽量推論向けのどちらなのかを判別するためのマッピング情報。FPGAへの設定パラメータ(出力フレームサイズ)の値でどちらかを判断しているため、設定パラメータ値と対応する推論タイプを記載している。

functionnamemap.json	説明	備考
{		
"sizeList":[{	高度推論用フィルタリサイズ用の情報	高度推論用
"height": 1280,	高度推論用フィルタリサイズの出力フレームの高さのサイズ	固定值
"width": 1280,	高度推論用フィルタリサイズの出力フレームの幅のサイズ	固定值
"functionName":"-high-infer"	高度推論用フィルタリサイズに該当するWBFunctionの名前に付け	る文字列
}.{	軽量推論用フィルタリサイズ用の情報	軽量推論用
"height": 416,	軽量推論用フィルタリサイズの出力フレームの高さのサイズ	固定值
"width": 416,	軽量推論用フィルタリサイズの出力フレームの幅のサイズ	固定值
"functionName":"-low-infer"	軽量推論用フィルタリサイズに該当するWBFunctionの名前に付け	る文字列
}]		
}		

タイプ別情報とファイル名のマッピング情報 ※自動収集&CM作成機能が内部で使用する、各種タイプ別情報とそのファイル名を結びつける情報。

premadefilelist.json	説明	備考
{		
"region-unique-info" : "region-unique-info.json",	「領域固有情報」とそのファイル名	固定値
"function-unique-info" : "function-unique-info.json",	「Function固有情報-共通属性」とそのファイル名	固定値
"filter-resize-childbs" : "filter-resize-childbs.json",	「Function固有情報-専用属性(filter/resize用)」とそのファイル名	固定値
}		

# 以下のコンフィグ情報ファイルは環境に合わせた変更が不要でそのまま使える。

**GPUFunc用コンフィグ情報** ※高度推論用(gpufunc-config-high-infer.json)と軽量推論用(gpufunc-config-low-infer.json)の2種類のコンフィグ情報を用意している。これら以外のGPUFunctionを使う場合は作成が必要。

# GPUFuntion用推論処理モジュール(高度推論を実施)のコンフィグ情報

GPUFuntion用推編処理センユール(高度推編を実施)のコンノイク情報 gpufunc-config-high-infer.json	説明	備考 (どの値も基本変更不要)
[{	入力側がDMA通信、出力側がRTP通信を行う場合の設定情報	
"rxProtocol": "DMA",	入力側のプロトコル	
"txProtocol":"RTP",	出力側のプロトコル	
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり
"imageURI": "localhost/localhost/qpu infer dma:1.1.0",	使用するコンテナのコンテナイメージ名	高度推論を実施するGPUFunc用推論処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無 ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は2nd NIC利用を前提としているため値は"true"固定 現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"virtualNetworkDeviceDriverType": "sriov", "envs":{	クアングションかどnd NLCで利用するfixt8NWデバイス用トライバ 使用するコンテナに設定する環境変数群	ったれる。STOV のかを削促さいている。数と値は STOV 回走 高度推論を実施するGPUFunc用推論処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"CUDA MPS PIPE DIRECTORY": "/tmp/nvidia-mps",	MPS機能間での相互通信用のディレクトリのフルバス	同項注意がも大売りあらい。Uniternitesiaを注しフェールのコンナナ大力に必要や外外を放く、pournitesia。Pospec.containersin,envicaxとされる
"CUDA_MPS_LOG_DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルバス。	
	PCIe接続において使用するDPDKで、アプリの起動方法(共有メモリの管理モード)を制御	# SATE WAS COMPANY OF THE PROPERTY OF THE PROP
"SHMEM_SECONDARY": "1",	するための情報。	推論アプリ単独で管理するプライマリモードは"0"を、PCIe接続のコントローラと連携管理するセカングリモードは"1"を設定する。基本"1"を設定する。
"HEIGHT": "1280",	入力映像のフレームサイズ(高さ)。	高度推論なので1280
"WIDTH": "1280"	入力映像のの入力フレームサイズ(幅)。	高度推論なので1280
},		as = Lossestat Telegraphy VIII of Lossest Lossest VIII of VIII
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1", "kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "qfunc-hi-1",	TOO CRESS 9 SOLD 7 TICKS 9 SIXX CITY IX	起動するコンテナは1台なので固定で良い
"workingDir": "/opt/nvidia/deepstream/deepstream-7.0",	コンテナのワーキングディレクトリ	値は"/opt/nvidia/deepstream/deepstream-X.Y"。"X.Y"の部分は使うdeepstreamのバージョン番号(基本変更不要)
"command": ["sh", "-c"],		実行コマンドの実態は"args"で指定している
"video/x-raw,format=(string)BGR,%WIDTH%,%BHEIGHT%", "I novideoconver! 'video/x-raw(memoyr:NVMM), format=(string)RGBA", "I m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1", "%WIDTH%", "%WIDTH%", "%HEIGHT%", "I queue I nvinfer config-file-path=./config_infer_primary_yoloV4_p6_th020_040.txt batch-size=1", "model-engine-file=/model_b1_gpu0_fp16.engine! queue! nvdsosd process-mode=1! nvvideoconvert!", "video/x-raw, format=(string)BGR'! videoconvert! queue! perf!rtpvrawpay! udpsink", "%GOUTPUTTPP%", "%GOUTPUTTPP%T," "%GOUTPUTTPORT96", "Sync=true"],	コマンド実行時に渡す引数	コンテナで実行する高度処理モジュールを軽量推論で実施するためのGstremerプラヴィンの実行コマンドと引数
"securityContext":{		
"securityContext":{     "privileged": true		值は"true"固定
"privileged": true },	lifecycleのフッカに関する鉛定(ハンドラー)を定義する	值は"true"固定
"privileged": true }, "lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する コンテナが終了する店前のフック、コンテナ終了店前に実施する内容を定義する	值は"true"固定
"privileged": true },	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"privileged": true }, "lifecycle":{ "preStop":{		値は"true"固定 コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定
"privileged": true }; "lifecycle":{     "preStop":{     "exec":{         "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{	コンテナが終了する直前のフック、コンテナ終了直前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定
"privileged": true }, "lifecycle":{  "preStop":{  "exec":{  "cxec":{  "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},  "volumeMounts":[{  "name": "hugepage-1qi",	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"Jで固定 起動するコンテナは1台なので固定で良い
"privileged": true }; "lifecycle":{     "preStop":{     "exec":{         "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":[{	コンテナが終了する直前のフック、コンテナ終了直前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定
"privileged": true }; "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "hugepage-1qi",         "mountPath": "/dev/hugepages" }; };	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"Jで固定 起動するコンテナは1台なので固定で良い
"privileged": true }; "lifecycle":{    "preStop":{    "exee":{    "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},    "volumeMounts": {{     "name": "hucepage-1qi",     "mountPath": "/dev/hugepages" };    "name": "host-nvidia-mps",	コンテナが終了する直前のフック、コンテナ終了直前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定
"privileged": true }; "lifecycle":{    "preStop":{    "exee":{    "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},    "volumeMounts":{{     "name": "huqepaqe-1qi",     "mountPath": "/dev/huqepaqes" };	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"Jで固定 起動するコンデナは1台なので固定で良い
"privileged": true };  "lifecycle":{     "preStop":{     "exec":{     "exec":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "husepage-1gl",         "mountPath": "/dev/hugepages"     };     // "name": "host-nvidia-mps",     "mountPath": '/tmp/nvidia-mps" };     // "name": "dpdk",	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
"privileged": true }; "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "hugepage-1qi",         "mountPath": "/dev/hugepages"         };         "name": "host-nvidia-mps",         "mountPath": "/rmp/nvidia-mps" };         "name": "dpdk",         "name": "dpdk",         "mountPath": "/var/run/dpdk"	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定
"privileged": true }, "ilfecycle":{  "preStop":{  "exec":{  "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},  "volumeMounts":{{  "name": "hugepage-1qi",  "mountPath": "/dev/hugepages"  },{  "name": "host-nvidia-mps",  "mountPath": "/tmp/nvidia-mps"  },{  "name": "dpdk",  "mountPath": "/var/run/dpdk"  }),	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
"privileged": true }; "lifecycle":{    "preStop":{    "exee":{    "command": ["sh","-c", "kill -KILL \${pidof gst-launch-1.0}"]}}},    "volumeMounts":[{         "name": "huqepaqe-10i",         "mountPath": "/dev/huqepaqes" };     "name": "host-nvidia-mps",    "mountPath": "/tmp/nvidia-mps" };     "name": "dpdk",    "name": "dpdk",    "name": "/yar/run/dpdk" };    "resources":{	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
"privileged": true }  // "iffecycle":{     "preStop":{     "exee":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "huqepage-1qi",         "mountPath": "/dev/huqepages" };{         "name": "host-nvidia-mps",         "mountPath": "/tmp/nvidia-mps" };      "name": "dpdk",     "name": "dpdk",     "mountPath": "/var/run/dpdk" };	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定 共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけないに対応するため設定。
"privileged": true }; "ilfecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "hugepage-1qi",         "mountPath": "/dev/hugepages"         };         "name": "host-nvidia-mps",         "mountPath": "/mp/nvidia-mps" };  **  **  **  **  **  **  **  **  **	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA MPS_PIPE_DIRECTORY"と同じ値
"privileged": true },  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", 'kill -KILL.\${pidof gst-launch-1.0}"]}}},  "volumeMounts":[{     "name": "huaepaae-1qi",     "mountPath": "/dev/huaepaaes" },{     "name": "host-nvidia-mps",     "mountPath": "/kmp/nvidia-mps" },{     "name": "dpdk",     "name": "dpdk",     "mountPath": "/var/run/dpdk" }],     "resources":{     "requests":{     "memory": "32Gi" },	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定 共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけないに対応するため設定。
"privileged": true }; "lifecycle":{    "preStop":{    "exee":{    "exee":{    "command": ["sh","-c", "kill -KILL \${pidof gst-launch-1.0}"]}}},    "volumeMounts":[{    "name": "hudepage-1al",    "mountPath": "/dev/hudepages" }; {    "name": "host-nvidia-mps",    "mountPath": "/tmp/nvidia-mps" };    "name": "dpdk",    "name": "dpdk",    "mountPath": "/var/run/dpdk" }];    "resources":{    "requests":{    "memory": "32Gi" },    "ilmits":{	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true },  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", 'kill -KILL.\${pidof gst-launch-1.0}"]}}},  "volumeMounts":[{     "name": "huaepaae-1qi",     "mountPath": "/dev/huaepaaes" },{     "name": "host-nvidia-mps",     "mountPath": "/kmp/nvidia-mps" },{     "name": "dpdk",     "name": "dpdk",     "mountPath": "/var/run/dpdk" }],     "resources":{     "requests":{     "memory": "32Gi" },	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepages"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定 共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけないに対応するため設定。
"privileged": true }; "lifecycle":{     "preStop":{     "exee":{         "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":[{         "name": "hucepaqe-1qi",         "mountPath": "/dev/hucepaqes" }; {         "name": "host-nvidia-mps",         "mountPath": "/tmp/nvidia-mps" };          "name": "dpdk",         "mountPath": "/var/run/dpdk" }];         "resources":{         "requests":{         "memory": "32Gi" },         "limits":{         "hucepaqes-1Gi": "1Gi" }	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低ど55か1つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true },  "lifecycle":{     "preStop":{         "exec":{         "command": ["sh","-c", "kill - KILL \${pidof gst-launch-1.0}"]}}},  "volumeMounts":{{         "name": "hucepace-1qi",         "mountPath": "/dev/hucepaqes" },  "name": "host-nvidia-mps",         "name": "host-nvidia-mps" },  "name": "dpdk",         "mountPath": "/var/run/dpdk" },  "resources":{         "requests":{         "memory": "32Gi" },         "limits":{         "huqepaqes-1Gi": "1Gi" }, }	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true }; "lifecycle":{     "preStop":{         "exec*:{         "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},     "volumeMounts":{{         "name": "hudepage-1al",         "mountPath": "/dev/hudepages"         },         "name": "host-nvidia-mps",         "mountPath": "/tmp/nvidia-mps"         },         "name": "dpdk",         "mountPath": "/var/run/dpdk"         }],         "resources":{         "requests":{         "memory": "32Gi"         },         "hudepages-1Gi": "1Gi"         }     },     "volumes":{	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepages"固定 環境変数"CUDA MPS PIPE DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低ど55か 1 つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true },  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", 'kill -KILL.\${pidof gst-launch-1.0}"]}}},  "volumeMounts":[{     "name": "huacepace-1qi",     "mountPath": "/dev/huacpaqes" },{     "name": "host-nvidia-mps",     "mountPath": "/kmp/nvidia-mps" },{     "name": "dpdk",     "mountPath": "/var/run/dpdk" }],     "resources":{     "requests":{     "memory": "32Gi" },     "limits":{     "huacpaqes-1Gi": "1Gi" } },  "volumes":[{     "name": "huacpage-1gi",	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互適信するためのディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低ど55か1つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true };  "lifecycle":{     "preStop":{         "exec":{         "command: ["sh","-c", "kill -KILL \${pidof gst-launch-1.0}"]}}},  "volumeMounts":[{         "name": "huaepaae-1ai",         "mountPath": "/dev/huaepaaes" };  "mountPath": "/dev/huaepaaes" };  "name": "host-nvidia-mps",         "mountPath": "/tmp/nvidia-mps" };  "mountPath": "/var/run/dpdk" };  "name": "dpdk",         "mountPath": "/var/run/dpdk" };  "resources":{         "reauests":{         "memory": "32Gi" },         "limits":{         "huaepaaes-1Gi": "1Gi" } };  "volumes":[{         "volumes":[{         "name": "hugepage-1gi",         "hostPath":	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか1つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true },  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},  "olumeMounts": [{     "name": "huqeaqe-1qi",     "mountPath": "/dev/huqepaqes" },{     "name": "host-nvidia-mps",     "mountPath": "/tmp/nvidia-mps" },{     "name": "hode",     "mountPath": "/var/run/dpdk" }],     "resources":{     "requests":{     "memory": "32Gi" },     "limits":{     "huqepaqes-1Gi": "1Gi" } },  "volumes":[{     "name": "huqepaqe-1qi",	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepages"固定 環境変数"CUDA MPS PIPE DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低ど55か 1 つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true };  "lifecycle":{     "preStop":{         "exec":{         "command": ["sh","-c", "kill -KILL \${pidof gst-launch-1.0}"]}}},  "volumeMounts":[         "name": "huaepade-1di",         "mountPath": "/dev/huaepades" };  /*  "name": "host-nvidia-mps",         "mountPath": "/tmp/nvidia-mps" };  "name": "oddk",         "mountPath": "/var/run/dpdk" };  "resources":{         "reduests":{         "memory": "32Gi" },         "limits":{         "huaepades-1Gi": "1Gi" } },  "volumes":[{         "name": "hugepade-1gi",         "nostPath": "/dev/huaepades"} };  ("path": "/dev/huaepades") };	コンテナが終了する値前のアック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepaqesディレクトリ用のVolumeMount MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA MPS PIPE DIRECTORY"と同じ値  値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけない」に対応するため設定。値は生意で良いので"32G"固定で良い 値は"1G"固定  上記volumeMounts."hugepage-1gi"と同じ値  上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値
"privileged": true },  "lifecycle":{  "preStop":{  "exee":{  "command": ["sh","-c", "kill - KILL \${pidof gst-launch-1.0}"]}}},  "volumeMounts": [{  "name": "huaepaae-1ai",  "mountPath": "/dev/huaepaaes"  },{  "name": "hust-ividia-mps",  "mountPath": "/tmp/nvidia-mps"  },{  "name": "host-ividia-mps",  "mountPath": "/var/run/dpdk"  }],  "resources":{  "requests":{  "requests":{  "memory": "32Gi"  },  "limits":{  "huaepaaes-1Gi": "1Gi"  } },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  }  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  }  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  }  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  },  "tolumes":[  "huaepaaes-1Gi": "1Gi"  hostPath":  "huaepaaes-1Gi": "host-nvidia-mps",	コンテナが終了する値前のフック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うハンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepaqesディレクトリ用のVolumeMount MPS用、MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値 値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか1つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い
"privileged": true };  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL.\$(pidof gst-launch-1.0)"]}}},     "volumedNounts":{{         "name": "husepage-1di",         "mountPath": "/dev/hugepages" };         "name": "host-nvidia-mps"         "name": "dodk",         "name": "dodk",         "mountPath": "/var/run/dpdk" }},     "resources":{         "requests":{         "memory": "32Gi" };     "limits":{         "hugepage-1gi",         "hostPath": "hugepage-1gi",         "hostPath": "hugepage-1gi",         "hostPath": "/dev/hugepages"} }; };	コンテナが終了する値前のアック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepaqesディレクトリ用のVolumeMount MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"Jで固定 起動するコンテナは1台なので固定で良い 値は"/dev/huqepaqes"固定 環境変数"CUDA MPS PIPE DIRECTORY"と同じ値  値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか1つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32G"固定で良い 値は"1G"固定  上記volumeMounts."hugepage-1gi"と同じ値  上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値
"privileged": true };  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL \${pidof gst-launch-1.0}"]}}},     "volumedwounts":{{         "name": "husepage-1ql",         "mountPath": "/dev/hugepages" };      "name": "host-nvidia-mps",     "name": "dodk",     "mountPath": "/var/run/dpdk" }},     "resources":{     "requests":{         "memory": "32Gi" };  };  "hudepages-1Gi": "1Gi" } };  "volumes":[     "name": "husepage-1gl",     "hostPath":         {"path": "fost-nvidia-mps",         "hostPath":         {"path": "fost-nvidia-mps",	コンテナが終了する値前のアック、コンテナ終了値前に実施する内容を定義する preStop時にコマンド実行を行うルンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/hugepaqesディレクトリ用のVolumeMount MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台2ので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA MPS PIPE DIRECTORY"と同じ値  値は"var-run-dpdk"固定  共有メモリ(hugepage)用, k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか 1 つ)をリクエストしなければいけない」に対応するため設定。値は任意で良いので"32Gi"固定で良い  値は"1Gi"固定  上記volumeMounts."hugepage-1gi"と同じ値 上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値 上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値
"privileged": true },  "lifecycle":{     "preStop":{     "exec":{     "command": ["sh","-c", "kill -KILL.\$(pidof gst-launch-1.0)"]}}},  "volumeMounts":{{     "name": "husepage-1d",     "mountPath": "/dev/husepages" }, {     "name": "host-nvidia-mps",     "mountPath": "/rar/run/dpdk" }},  "name": "dpdk",     "mountPath": "/var/run/dpdk" }},  "resources":{     "resources":{     "remests":{     "memony": "32Gi" },     "limits":{     "husepage-1Gi": "1Gi" } },  "hostPath": "/dev/husepages"} },{     "name": "host-nvidia-mps",     "hostPath":     "path": "/foev/husepages"} },  "name": "host-nvidia-mps",     "hostPath":     ""ame": "host-nvidia-mps",     "hostPath":     ""ame": "host-nvidia-mps",     "hostPath":     ""ame": "host-nvidia-mps",     "hostPath":     ""path": "/tmp/nvidia-mps",     "hostPath":     ""path": "/tmp/nvidia-mps") },{	コンテナが終了する値前のアック、コンテナ終了値前に実施する内容を定義する prestoo時にコント実行を行うパンドラー Pod停止用コマンド 入力側のPCIe接続で使用する/dev/huqepagesディレクトリ用のVolumeMount  MPS用、MPS機能間で相互通信するためのディレクトリ用のVolumeMount 入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount  入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount  MPS用、MPS機能間で相互通信するためのディレクトリ用のVolume  MPS用、MPS機能間で相互通信するためのディレクトリ用のVolume	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定 起動するコンテナは1台なので固定で良い 値は"/dev/hugepages"固定 環境変数"CUDA MPS_PIPE_DIRECTORY"と同じ値  値は"var-run-dpdk"固定  共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか1つ)をリクエストしなければいけないに対応するため設定。値は任意で良いので"32Gi"固定で良い  値は"1Gi"固定  上記volumeMounts."hugepage-1gi"と同じ値 上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値 上記volumeMounts."host-nvidia-mps"と同じ値

22	T	
}], "hostNetwork": false.		現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostNetwork": false, "hostIPC": true,		がれるとHU NICやサガを用り定としているだめ間は Taise CRUs。
"restartPolicy": "Always"		
	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	値は"true"固定
"shareProcessNamespace": true	rouryのコンテナinでのプロで人有別上inの共有で有別にするが音がの設定	Bello ci un Bello.
1		
1		
r r	入力側がTCP通信、出力側がRTP通信を行う場合の設定情報	
"rxProtocol": "TCP",	入力側のプロトコル	
"txProtocol":"RTP",	出力側のプロトコル	
"imageURI": "localhost/qpu infer tcp:1.1.0",	使用するコンテナのコンテナイメージ名	高度推論を実施するGPUFunc用推論処理モジュールのコンテナイメージ名。タグの数字はパージョン番号(基本変更不要)
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	同及推論を実施するBrurun在語文壁モジュールのコンティイメージ者。 ダクの数子はパージョン音号(基本変更不安) 現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
	ファンクションかとnd NiL で利用するNXはNWテハイス用トフィハ 体田ナスランニナバールウナスで見合なを記さ	
"envs":{	使用するコンテナに設定する環境変数群	高度推論を実施するGPUFunc用推論処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"CUDA_MPS_PIPE_DIRECTORY": "/tmp/nvidia-mps",	MPS機能間での相互通信用のディレクトリのフルバス	
"CUDA_MPS_LOG_DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルバス。	
"GST_PLUGIN_PATH": "/opt/nvidia/deepstream/deepstream-7.0/sample-	コンテナ内でのGstreamerプラグインの格納ディレクトリ。	
functions/functions/gpu_infer_tcp_plugins/fpga_depayloader",		
"HEIGHT": "1280",	入力映像のフレームサイズ(高さ)。	高度推論なので1280
"WIDTH": "1280"	入力映像のの入力フレームサイズ(幅)。	高度推論なので1280
},		
"template":{		
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "gfunc-hi-1",		起動するコンテナは1台なので固定で良い
"workingDir": "/opt/nvidia/deepstream/deepstream-7.0",	コンテナのワーキングディレクトリ	値は"/opt/nvidia/deepstream/deepstream-X.Y"。"X.Y"の部分は使うdeepstreamのバージョン番号(基本変更不要)
"command": ["sh", "-c"],		実行コマンドの実態は"args"で指定している
"args":["cd /opt/DeepStream-Yolo && gst-launch-1.0 -ev fpgadepay",		The Constant Fig. Constant
"%INPUTIP%",		
"%INPUTPORT%",		
"! 'video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%'",		
"! nvvideoconvert ! 'video/x-raw(memory:NVMM), format=(string)RGBA'",		
"! m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1",		
"%WIDTH%",		
"%HEIGHT%",	コマンド実行時に渡す引数	コンテナで実行する高度処理モジュールを軽量推論で実施するためのGstremerプラグインの実行コマンドと引数
"! queue ! nvinfer config-file-path=./config_infer_primary_yoloV4_p6_th020_040.txt batch-size=1",	コインド矢打時に成り引致	コンテナ(矢1」9つ同及22年モンユールで発星作調(矢応9つにの2005tieffler)プソインの矢1リコインドと引致
"model-engine-file=./model_b1_gpu0_fp16.engine ! queue ! nvdsosd process-mode=1 ! nvvideoconvert		
I'',		
"'video/x-raw, format=(string)BGR'! videoconvert! queue! perf!rtpvrawpay! udpsink", "%OUTPUTIP%",		
"%OUTPUTPORT%",		
"sync=true"],		
"securityContext":{		
"privileged": true		
},	100 L 67 51698478057115 157 345547	
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{	preStop時にコマンド実行を行うハンドラー	
"command": ["sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"]}}},	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定
"volumeMounts":[{		
"name": "host-nvidia-mps",	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount	
"mountPath": "/tmp/nvidia-mps"		環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
}]		
Y],		
"volumes":[{		
"name": "host-nvidia-mps",	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolume	上記volumeMounts."host-nvidia-mps"と同じ
"hostPath":	The second second second 19 of	
{"path": "/tmp/nvidia-mps"}		上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値
31.		The state of the s
"hostNetwork": false,		現状は2nd NIC利用を前提としているため値は"false"で良い。
		ANAMERIA INTOTYTI COMPECO CANACOMENIA TODE CIXAN
"hostIPC": true, "restartPolicy": "Always"		
	D. Johns こと887のプロセスク 熱か88の共存と存むにせてもている****	族(+1) por 10 (中) 中
"shareProcessNamespace": true	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	值は"true"固定
}		
<u> </u>		
[}]		

# GPUFuntion用推論処理モジュール(軽量推論を実施)のコンフィグ情報

gpufunc-config-low-infer.json	説明	備考(どの値も基本変更不要)
[{	入力側がDMA通信、出力側がRTP通信を行う場合の設定情報	
"rxProtocol": "DMA",	入力側のプロトコル	
"txProtocol":"RTP",	出力側のプロトコル	
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり
"imageURI": "localhost/gpu infer dma:1.1.0",	使用するコンテナのコンテナイメージ名	軽量推論を実施するGPUFunc用推論処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"envs":{	使用するコンテナに設定する環境変数群	軽量推論を実施するGPUFunc用推論処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"CUDA MPS PIPE DIRECTORY": "/tmp/nvidia-mps",	MPS機能間での相互通信用のディレクトリのフルバス	
"CUDA_MPS_LOG_DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルバス。	

IICHMEM CECONDADVIII. II4II	DOI-104(-1-1), 7 (4 D + 7 D D D ) 7 (2 D 1 D 1 D 1 D 1 D 1 D 1 D 1 D 1 D 1 D	#
"SHMEM_SECONDARY": "1", "HEIGHT": "416",	PCIe接続において使用するDPDKで、アフリの起動方法(共有メモリの管理モード)を制御入力映像のフレームサイズ(高さ)。	# 推論アプリ単独で管理するプライマリモードは"0"を、PCIe接続のコントローラと連携管理するセカンダリモードは"1"を設定する。基本"1"を設定する。 軽量推論なので416
"WIDTH": "416"	入力映像のの入力フレームサイズ(幅)。 入力映像のの入力フレームサイズ(幅)。	軽量推論なので416
},	7 Open and the Property of the	
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{  "containers":[{	Podで起動するコンテナに関する設定情報	
"name": "gfunc-n02-lo-1",	POUC起動するコンナルに関する政定情報	起動するコンテナは1台なので固定で良い
"workingDir": "/opt/nvidia/deepstream/deepstream-7.0",	コンテナのワーキングディレクトリ	値は"/opt/nvidia/deepstream/deepstream-X.Y"。"X.Y"の部分は使うdeepstreamのパージョン番号(基本変更不要)
"command": ["sh", "-c"],	使用するコンテナでの実行コマンド	実行コマンドの実態は"args"で指定している
"args":["cd /opt/nvidia/deepstream/deepstream-7.0/sources/objectDetector_Yolo/ && gst-launch-1.0 -er fpgasrc!",  "video/x-raw,format=(string)BGR, 9sWIDTH9, %HEIGHT16",  "I nvideoconvert! Video/x-raw(memory:NVMM), format=(string)RGBA",  "I m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-isze=1",  "9sWIDTH98",  "9sWIDTH98",  "9sWIEGHT96",  "1 queue ! nvinfer config-file-path=/config_infer_primary_yoloV3_tiny.txt",  "batch-isze=1 model-engine-file=_/model_b1_gpu0_int8.engine! queue! nvivideoconvert!",  "video/x-raw, format=(string)BGR'! videoconvert! queue! perf! rtpvrawpay! udpsink",  "9sCUTPUTP98",  "9sCUTPUTPORT%",  "9scuttyContext":{	マコマンド実行時に渡す引数	コンテナで実行する推論処理モジュールを軽量推論で実施するためのGstremerプラグインの実行コマンドと引数
"privileged": true		值は"true"固定
},		
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{     "command": ["sh","-c", "kill -KILL \$(pidof qst-launch-1.0)"]}}},	preStop時にコマンド実行を行うハンドラー Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -KILL \$(pidof gst-launch-1.0)"」で固定
"volumeMounts":[{		,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
"name": "huqepaqe-1qi",	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	値は"huqepaqe-1qi"固定
"mountPath": "/dev/hugepages"		值は"/dev/hugepages"固定
}.{	MANUAL MANUAL PROPERTY OF THE PARTY OF THE P	
"name": "host-nvidia-mps", "mountPath": "/tmp/nvidia-mps"	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount	環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
},{		SANDERA GOOTEN OF A CONTROL OF CHICAGO
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	
"mountPath": "/var/run/dpdk"		值は"var-run-dpdk"固定
}],		
"resources":{     "requests":{		
"memory": "32Gi"		共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低ど5らか1つ)をリクエストしなければいけない」に対応するため設定。 値は任意で良いので"32GI"固定で良い
}, "limits":{		
"hugepages-1Gi": "1Gi"	hugepageの1枚分のページサイズ	值は"1Gi"固定
}	Hudebadeo 14XXXVIII VIII	MICHAEL A. LANGE AND
}		
}],		
"volumes":[{	1 + Planct + 校结支持 田 + 7 / / /	h52.clumeMounts "hugopage 1si"kEll" iti
"name": "hugepage-1gi", "hostPath":	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolume	上記volumeMounts."hugepage-1gi"と同じ値
{"path": "/dev/hugepages"}		上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値
},{		
"name": "host-nvidia-mps",	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolume	上記volumeMounts."host-nvidia-mps"と同じ値
"hostPath":		上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値
{"path": "/tmp/nvidia-mps"} },{		⊒aground tourts, nost rivide mps of mount dut c ∞jO M
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolume	上記volumeMounts."dpdk"と同じ値
"hostPath":		
{"path": "/var/run/dpdk"}		上記volumeMounts."dpdk"の"mountPath"と同じ値
}], "hostNetwork": false,		現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostIPC": true,		がAnd 2 Hu NIC 利用を削煙ことでいる人が組まればいません。 値は"true"固定
"restartPolicy": "Always"		值は"Always"固定
"shareProcessNamespace": true	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	值は"true"固定
}		
<u>}</u>		
<u></u>	入力側がTCP通信、出力側がRTP通信を行う場合の設定情報	
"rxProtocol": "TCP",	入力側のプロトコル	
"txProtocol":"RTP",	出力側のプロトコル	
	休田オスコンニナのコンニナノノーミク	軽量推論を実施するGPUFunc用推論処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"imageURI": "localhost/gpu_infer_tcp:1.1.0",	使用するコンテナのコンテナイメージ名	
"imageURI": "localhost/gpu_infer_tcp:1.1.0",  "additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定 現状は2nd NIC利用を前提としているため値は"true"固定
"imageURI": "localhost/gpu_infer_tcp:1.1.0",	ファンクションの2nd NICの利用の有無 ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"imageURI": "localhost/apu_infer_tcp:1.1.0", "additionalNetwork": true, "virtualNetworkDeviceDriverType": "sriov",	ファンクションの2nd NICの利用の有無	

"GST_PLUGIN_PATH": "/opt/nvidia/deepstream/deepstream-7.0/sample-	コンテナ内でのGstreamerブラグインの格納ディレクトリ。	
functions/functions/gpu_infer_tcp_plugins/fpga_depayloader",		TREMONORMS
"HEIGHT": "416", "WIDTH": "416"	入力映像のフレームサイズ(高さ)。	軽量推論なので416 軽量推論なので416
"WIDTH": "416"	入力映像のの入力フレームサイズ(幅)。	整悪推論なので416
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1",	TFDX.9 つPOUのナンフレードナータ	コンナナにaxたyの外外を数(env)はアンプレートのか下部(工)にいまれてしている
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "gfunc-n02-lo-1",	TOO CALEDY Y BLOW Y BUCKLINTIK	起動するコンテナは1台なので固定で良い
"workingDir": "/opt/nvidia/deepstream/deepstream-7.0",	コンテナのワーキングディレクトリ	値は"/opt/nvidia/deepstream/deepstream-X.Y"。"X.Y"の部分は使うdeepstreamのバージョン番号(基本変更不要)
"command": ["sh", "-c"],	使用するコンテナでの実行コマンド	ま行うコンドの実際は"args"で指定している
"args":["cd /opt/nvidia/deepstream/deepstream-7.0/sources/objectDetector_Yolo/ && gst-launch-1.0 -ev		
ftpgadeay",  "\$INPUTIP96",  "\$INPUTIP96",  "\$INPUTIP96",  "I video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%",  "I nvideoconvert! video/x-raw(memory.NVMM), format=(string)RGBA",  "I m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1",  "%WIDTH%",  "%HEIGHT%",  "\$HEIGHT%",  "I queue I nvinfer config-file-path=/config_infer_primary_voloV3_tiny.txt",  "batch-size=1 model-engine-file=/model_b1_gpu0_int8.engine! queue! nvvideoconvert!",  "video/x-raw, format=(string)BGR'! videoconvert! queue! perf!rtpvrawpay! udpsink",  "%GOUTPUTIPOK",  "Synce-true"],	コマンド実行時に渡す引数	コンテナで実行する推論処理モジュールを軽量推論で実施するためのGstremerプラヴィンの実行コマンドと引数
"securityContext":{		
"privileged": true		
},		
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{     "command": ["sh","-c", "kill -KILL \$(pidof qst-launch-1.0)"]}}},	preStop時にコマンド実行を行うハンドラー Pod停止用コマンド	コマンドラインは「"sh", "-c", "kill -KILL \$(pidof qst-launch-1.0)"」で固定
"volumeMounts":[{	POOI字正用コイント	コインドノインは「SII,-C, KIII-KILL \$(pidoi gst-iadricit-1.0)]で副走
"name": "host-nvidia-mps",	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolumeMount	
"mountPath": "/tmp/nvidia-mps"	PIF3所。PIF3域形向(有互應百分のにあり)インフェッカのVolumeriounic	環境変数"CUDA MPS PIPE DIRECTORY"と同じ値
}]		3873030 GOOT_110_1110111_GOOT_110_10
χ1.		
"volumes":[{		
"name": "host-nvidia-mps",	MPS用。MPS機能間で相互通信するためのディレクトリ用のVolume	上記volumeMounts."host-nvidia-mps"と同じ
"hostPath":		The state of the s
{"path": "/tmp/nvidia-mps"}		上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値
}],		
"hostNetwork": false,		現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostIPC": true,		值は"true"固定
"restartPolicy": "Always"		值は"Always"固定
"shareProcessNamespace": true	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	值は"true"固定
}		
}		
}]		

# FPGAFunc用コンフィグ情報

※高度推論向けのフィルタリサイズ用(fpgafunc-config-filter-resize-high-infer,ison)と軽量推論向けのフィルタリサイズ用(fpgafunc-config-filter-resize-low-infer,ison)の2種類のコンフィグ情報を用意している。これら以外のFPGAFunctionを使う場合は作成が必要。 (フィルタリサイズ用が高度推論向けと軽量推論向けに分かれている様に)入カバラメータの値ごとに別途コンフィグ情報を作成する。(同じ処理でも入力バラメータの値が異なると別のFPGAFunctionとして定義する想定なため)

# FPGAFunc用フィルタリサイズ処理モジュール(高度推論向けの処理を実施)のコンフィグ情報

fpgafunc-config-filter-resize-high-infer.json	説明	備考(どの値も基本変更不要)
(		
"parentBitstream": {	使用する親Bitstreamの情報	
"file": "OpenKasugai-fpga-example-design-1.0.0-1.mcs",	親Bitstreamのファイル名	
"id": "0100001c"	親BitstreamのピットストリームID	
},		
"childBitstream": {	使用する子Bitstreamの情報	
"file": "OpenKasugai-fpga-example-design-1.0.0-2.bit",	子Bitstreamのファイル名	
"id": "0100001c"	子BitstreamのビットストリームID	
},		
"parameters": {	使用するFPGAの子bsに設定する環境変数群	高度推論用フィルタリサイズ処理モジュール用bitstreamに設定するバラメータ群
"functions": {	設定先のモジュール名	フィルタリサイズの場合はfunctionsモジュールにのみ処理モジュール用パラメータを設定すれば良い
"i_width": 3840,	入力フレームの幅のサイズ	高度用でも軽量用でも3840
"i_height": 2160,	入力フレームの高さのサイズ	高度用でも軽量用でも2160
"o_width": 1280,	出力フレームの幅のサイズ	高度推論用なので1280
"o_height": 1280	出力フレームの高さのサイズ	高度推論用なので1280
}		
},		
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり。(基本"256"で良い)
"functionDedicatedInfo": "filter-resize-ch"		
"functionName": "filter-resize-high-infer"	Functionの名称	高度用は"filter-resize-high-infer"
}		

FPGAFunc用フィルタリサイズ処理モジュール(軽量推論向けの処理を実施)のコンフィグ情報

fpgafunc-config-filter-resize-low-infer.json	説明	備考(どの値も基本変更不要)	
{			
"parentBitstream": {	使用する親Bitstreamの情報		
"file": "OpenKasugai-fpga-example-design-1.0.0-1.mcs",	親Bitstreamのファイル名		
"id": "0100001c"	親BitstreamのピットストリームID		
},			
"childBitstream": {	使用する子Bitstreamの情報		
"file": "OpenKasugai-fpga-example-design-1.0.0-2.bit",	子Bitstreamのファイル名		
"id": "0100001c"	子BitstreamのビットストリームID		
},			
"parameters": {	使用するFPGAの子bsに設定する環境変数群	軽量推論用フィルタリサイズ処理モジュール用bitstreamに設定するバラメータ群	
"functions": {	設定先のモジュール名	フィルタリサイズの場合はfunctionsモジュールにのみ処理モジュール用バラメータを設定すれば良い	
"i_width": 3840,	入力フレームの幅のサイズ	高度用でも軽量用でも3840	
"i_height": 2160,	入力フレームの高さのサイズ	高度用でも軽量用でも2160	
"o_width": 416,	出力フレームの幅のサイズ	軽量推論用なので416	
"o_height": 416	出力フレームの高さのサイズ	軽量推論用なので416	
}			
},			
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり。(基本"256"で良い)	
"functionDedicatedInfo": "filter-resize-ch"			
"functionName": "filter-resize-low-infer"	Functionの名称	軽量用は"filter-resize-low-infer"	
}			

# CPUFunc用コンフィグ情報

※デコード用(cpufunc-config-filter-resize-low-infer.json)と高度推論向けのフィルタリサイズ用(cpufunc-config-copy-branch.json)と Glue用(cpufunc-config-glue-fdma-to-tcp.json)の5種類のコンフィグ情報を用意している。これら以外のCPUFunctionを使う場合は作成が必要。 なお、入カパラメータの値ごとに別途コンフィグ情報を作成する。(入カパラメータの値が異なると別のCPUFunctionとして定義する想定なため)

## CPUFuntion用デコード処理モジュールのコンフィグ情報

cpufunc-config-decode.json	説明	備考(どの値も基本変更不要)
[{	入力側がRTP通信、出力側がDMA通信を行う場合の設定情報	
"rxProtocol":"RTP",	入力側のプロトコル	"RTP"固定
"txProtocol":"DMA",	出力側のプロトコル	"DMA"固定
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり。(基本"256"で良い)
"imageURI": "localhost/cpu_decode:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用デコード処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"envs":{	使用するコンテナに設定する環境変数群	CPUFunc用デコード処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"DECENV APPLOG LEVEL": "6",	ログレベル	値は特に変更不要
"DECENV FRAME WIDTH": "3840",	入力映像のフレームサイズ(幅)	サンブルユースケースでは3840固定
"DECENV FRAME HEIGHT": "2160",	入力映像のフレームサイズ(高さ)	サンブルユースケースでは2160固定
"DECENV VIDEO CONNECT LIMIT": "0",	映像送信元との連続接続回数。	デフォルト値"0"のままで良い
"DECENV VIDEOSRC PROTOCOL": "RTP",	受信するプロトコル	"RTP"固定
"DECENV OUTDST PROTOCOL": "DMA"	送信するプロトコル	"DMA"固定
},		
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "cfunc-1".	THE CALL OF THE CA	起動するコンテナは1台なので固定で良い
"command": ["sh","-c"],	使用するコンテナでの実行コマンド	実行コマンドの実態は"args"で指定している
"args":["./sample-functions/functions/cpu_decode/build/cpu_decode-shared"],	コマンド実行時に渡す引数	CPUFunc用デコード処理モジュールをコンテナで実行するためのコマンド
"securityContext":{		The state of the s
"privileged": true		值は"true"固定
}		Mark V V V party
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{	preStop時にコマンド実行を行うハンドラー	
"command": ["sh","-c", "kill -TERM \$(pidof cpu_decode-shared)"]}}},	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof cpu decode-shared)"」で固定
"volumeMounts":[{	1 0017111111111111111111111111111111111	1 2/2/2/200 3/7 C/ Nill File (\$1/2/2)
"name": "hugepage-1gi",	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	値は"hugepage-1gi"固定
"mountPath": "/dev/huqepaqes"	7(7) (6) OF CICIDANG CICINI S S/ dev/ Hagepages 7 (10 7) 3/100 volanie Hount	into rough of manage in m
} {		Bits / dc v/Huqcpaqc5 BitA_
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount	
"mountPath": "/var/run/dpdk"	7(7) jejori Cicigani, Cig/ij 9 abri bir Cig/ij 9 abri 10 ij 7/10 7/10 volumenoune	值は"var-run-dodk"固定
3]		Bate var run dpak sax.
"resources":{		
"requests":{		
requests :3		共有メモリ(hugepage)用。k8sの仕様「hugepageを使用する際には、CPUやメモリ(最低どちらか1つ)をリクエストしなければいけない」に対応するため設定。
"memory": "32Gi"		
		値は任意で良いので"32Gi"固定で良い
10		
"limits":{	h	值は"1G "固定
"hugepages-1Gi": "1Gi"	hugepageの1枚分のページサイズ	TELS TO DEC
}		
}		
}],		
"volumes":[{		1

"name": "hugepage-1qi",	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolume	上記volumeMounts."hugepage-1gi"と同じ値
"hostPath":	77730001 Cleigeweet XIII y by de ly magepages 7 (b 7 ) 7/100 formite	and the state of t
{"path": "/dev/hugepages"}		上記volumeMounts."hugepage-1qi"の"mountPath"と同じ値
73		and the state of t
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolume	上記volumeMounts."dpdk"と同じ値
"hostPath":	7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	
{"path": "/var/run/dpdk"}		上記volumeMounts,"dpdk"の"mountPath"と同じ値
11		
"hostNetwork": false,		現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostIPC": true,		值は"true"固定
"restartPolicy": "Always"		值は"Always"固定
}		
}.		
{	入力側がRTP通信、出力側がTCP通信を行う場合の設定情報	
"rxProtocol":"RTP",	受信側(RTP接続)のプロトコル	"RTP"固定
"txProtocol":"TCP",	送信側(Eth接続)のプロトコル	"TCP"固定
"imageURI": "localhost/cpu_decode:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用デコード処理モジュールのコンテナイメージ名。タグの数字はパージョン番号(基本変更不要)
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"envs":{	使用するコンテナに設定する環境変数群	CPUFunc用デコード処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"DECENV APPLOG LEVEL": "6",	ログレベル	値は特に変更不要
"DECENV FRAME WIDTH": "3840",	入力映像のフレームサイズ(幅)	サンブルユースケースでは3840固定
"DECENV FRAME HEIGHT": "2160",	入力映像のフレームサイズ(高さ)	サンブルユースケースでは2160固定
"DECENV VIDEO CONNECT LIMIT": "0",	映像送信元との連続接続回数。	デフォルト値"0"のままで良い
"DECENV VIDEOSRC PROTOCOL": "RTP",	受信するプロトコル	"RTP"固定
"DECENV OUTDST PROTOCOL": "TCP"	送信するプロトコル	"TCP"固定
},		
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "cfunc-1",		起動するコンテナは1台なので固定で良い
"command": ["sh","-c"],	使用するコンテナでの実行コマンド	実行コマンドの実態は"args"で指定している
"args":["./sample-functions/functions/cpu_decode/build/cpu_decode-shared"],	コマンド実行時に渡す引数	内容はCPUFunc用デコードの実行コマンド(実行ファイルのバス)
"securityContext":{		
"privileged": true		值は"true"固定
},		
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{	preStop時にコマンド実行を行うハンドラー	
"command": ["sh","-c", "kill -TERM \$(pidof cpu_decode-shared)"]}}}	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof cpu_decode-shared)"」で固定
}],		
"hostNetwork": false,	Dockerホスト側のネットワークスタックを使用するためのコンテナのネットワーク設定	現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostIPC": true,		值は"true"固定
"restartPolicy": "Always"		值は"Always"固定
}		
1		
\$		

フィルタリサイズ処理モジュール(高度推論向けの処理を実施)向けのCPUFuntion用コンフィグ情報

cpufunc-config-filter-resize-high-infer.json	説明	備考(どの値も基本変更不要)
{	入出力ともTCP通信を行う場合の設定情報	現状は入出力ともTCP通信向けにのみ対応しているので、この場合の設定情報しかない
"rxProtocol":"TCP",	入力側のプロトコル	"TCP"固定
"txProtocol": "TCP",	出力側のプロトコル	"TCP"固定
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"imageURI": "localhost/cpu_filter_resize:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用デコード処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"envs":{	使用するコンテナに設定する環境変数群	高度推論を実施するCPUFunc用推論処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"FRENV APPLOG LEVEL": "DEBUG",	ログレベル	値は特に変更不要
"FRENV_INPUT_WIDTH": "3840",	入力映像のフレームサイズ(幅)	サンブルユースケースでは3840固定
"FRENV_INPUT_HEIGHT": "2160",	入力映像のフレームサイズ(高さ)	サンブルユースケースでは2160固定
"FRENV OUTPUT WIDTH": "1280",	出力映像のフレームサイズ(幅)	高度推論なので1280
"FRENV_OUTPUT_HEIGHT": "1280"	出力映像のフレームサイズ(高さ)	高度推論なので1280
},		
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "fr",		起動するコンテナは1台なので固定で良い
"command": ["python",		
"fr.py",		
"in_port=\$(FRENV_INPUT_PORT)",		
"out_addr=\$(FRENV_OUTPUT_IP)",		
"out_port=\$(FRENV_OUTPUT_PORT)",	使用するコンテナでの実行コマンド	CPUFunc用フィルタリサイズ処理モジュールを高度推論向けに実施するためにコンテナで実行するためのコマンド
"in_width=\$(FRENV_INPUT_WIDTH)",	was a second of	
"in_height=\$(FRENV_INPUT_HEIGHT)",		
"out_width=\$(FRENV_OUTPUT_WIDTH)",		
"out_height=\$(FRENV_OUTPUT_HEIGHT)",		
"loglevel=\$(FRENV_APPLOG_LEVEL)"],		

"securityContext":{			
"privileged": true		値は"true"固定	
},			
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する		
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する		
"exec":{	preStop時にコマンド実行を行うハンドラー		
"command": ["sh","-c", "kill -TERM \$(pidof python)"]}}}	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof python)"」で固定	
}],			
"hostNetwork": false,	Dockerホスト側のネットワークスタックを使用するためのコンテナのネットワーク設定	現状は2nd NIC利用を前提としているため値は"false"で良い。	
"hostIPC": true,		値は"true"固定	
"restartPolicy": "Always"		值は"Always"固定	
"shareProcessNamespace": true	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	値は"true"固定	
}			
}			
31			·

フィルタリサイズ処理モジュール(軽量推論向けの処理を実施)向けのCPUFuntion用コンフィグ情報

cpufunc-config-filter-resize-low-infer.json	説明	備考(どの値も基本変更不要)
[{	入出力ともTCP通信を行う場合の設定情報	現状は入出力ともTCP通信向けにのみ対応しているので、この場合の設定情報しかない
"rxProtocol": "TCP",	入力側のプロトコル	"TCP"固定
"txProtocol": "TCP",	出力側のプロトコル	"TCP"固定
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"imageURI": "localhost/cpu_filter_resize:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用デコード処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"envs":{	使用するコンテナに設定する環境変数群	高度推論を実施するCPUFunc用推論処理モジュールのコンテナ実行に必要な環境変数で、pod用テンプレート配下のspec.containers[i].envに設定される
"FRENV APPLOG LEVEL": "DEBUG",	ログレベル	值は特に変更不要
"FRENV INPUT WIDTH": "3840",	入力映像のフレームサイズ(幅)	サンブルユースケースでは3840固定
"FRENV INPUT HEIGHT": "2160",	入力映像のフレームサイズ(高さ)	サンブルユースケースでは2160固定
"FRENV OUTPUT WIDTH": "416",	出力映像のフレームサイズ(幅)	高度推論なので1280
"FRENV OUTPUT HEIGHT": "416"	出力映像のフレームサイズ(高さ)	高度推論なので1280
t	EDJAKOV ZYTA(INC)	INDECTAMENT A TOO
"template":{	作成するPodのテンプレートデータ	コンテナに設定する環境変数(env)はテンプレートの外部(上述のenvs)に記載している
"apiVersion": "v1".	1F/0X,9*SFOU(0)) 27 D = [17] = 9	コンナンにBXX上りるがかをXX人(CITV)はアンファレートのプログ(これでのCITVS)にBDがたび(くいる
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "fr".	POUC起動するコンナバに対する政定情報	起動するコンテナは1台なので固定で良い
"command": ["python",		尼瀬がるコフナノは1日はのく国民ではない
"fr.pv",  "-in_port=\$(FRENV_INPUT_PORT)",  "-out_addr=\$(FRENV_OUTPUT_IP)",  "-out_port=\$(FRENV_OUTPUT_PORT)",  "-in_width=\$(FRENV_INPUT_WIDTH)",  "-in_beight=\$(FRENV_INPUT_HEIGHT)",  "-out_width=\$(FRENV_OUTPUT_HEIGHT)",  "-out_height=\$(FRENV_OUTPUT_HEIGHT)",  "-loidy=\$(FRENV_OUTPUT_HEIGHT)",	使用するコンテナでの実行コマンド	CPUFunc用フィルタリサイズ処理モジュールを軽量推論向けた実施するためにコンテナで実行するためのコマンド
"securityContext":{		
"privileged": true		值は"true"固定
<u>},</u>		
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する	
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する	
"exec":{	preStop時にコマンド実行を行うハンドラー	
"command": ["sh","-c", "kill -TERM \$(pidof python)"]}}}	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof python)"」で固定
}],		
"hostNetwork": false,	Dockerホスト側のネットワークスタックを使用するためのコンテナのネットワーク設定	現状は2nd NIC利用を前提としているため値は"false"で良い。
"hostIPC": true,		值は"true"固定
"restartPolicy": "Always"		值は"Always"固定
"shareProcessNamespace": true	Pod内のコンテナ間でのプロセス名前空間の共有を有効にするか否かの設定	值は"true"固定
}		
}		
\1		

コピー分岐処理モジュール向けのCPUFuntion用コンフィグ情報

cpufunc-config-copy-branch.json	説明	備考(どの値も基本変更不要)
[{	入出力ともTCP通信を行う場合の設定情報	現状は入出力ともTCP通信向けにのみ対応しているので、この場合の設定情報しかない
"rxProtocol":"TCP",	入力側のプロトコル	"TCP"固定
"txProtocol": "TCP",	出力側のプロトコル	"TCP"固定
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前提としているため、設定値は"sriov"固定
"copyMemorySize": "1024",	TCP受信データ格納メモリ1領域あたりのメモリサイズ情報(Byte)	デフォルト値"1024"固定
"imageURI": "localhost/cpu copy branch:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用コピー分岐処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)
"template":{	作成するPodのテンプレートデータ	
"apiVersion": "v1",		
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "cfunc-copy-branch-1",		起動するコンテナは1台なので固定で良い
"workingDir": "/opt/openkasugai-controller/sample-functions/functions-ext/cpu copy branch",	コンテナのワーキングディレクトリ	値は"/opt/openkasugai-controller/sample-functions/functions-ext/cpu copy branch"固定
"command": ["sh","-c"],	使用するコンテナでの実行コマンド	実行コマンドの実態は"args"で指定している

"args":[",/copy_branch",	コマンド実行時に渡す引数	CPUFunc用コピー分岐処理モジュールの実行コマンドとその引数	
"securityContext":{			
"privileged": true		值は"true"固定	
},			
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する		
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する		
"exec":{	preStop時にコマンド実行を行うハンドラー		
"command": ["sh","-c", "kill -TERM \$(pidof copy_branch)"]}}}	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof copy_branch)"」で固定	
}],			
"hostNetwork": false,	Dockerホスト側のネットワークスタックを使用するためのコンテナのネットワーク設定	現状は2nd NIC利用を前提としているため値は"false"で良い。	
"hostIPC": true,		值は"true"固定	
"restartPolicy": "Always"		值は"Always"固定	
}			
}			
}]			

qlue(DMA→TCP変換)処理モジュール向けのCPUFuntion用コンフィグ情報

	lue(DMA→TCP変換)処理モジュール向けのCPUFuntion用コンフィグ情報			
cpufunc-config-glue-fdma-to-tcp.json	説明	備考(どの値も基本変更不要)		
Iţ.	入力側がDMA通信、出力側がTCP通信を行う場合の設定情報	DMA→TCPの変換用の処理モジュールなのでこの場合の設定情報しかない		
"rxProtocol":"DMA",	入力側のプロトコル	"DMA"固定		
"txProtocol":"TCP",	出力側のプロトコル	"TCP"固定		
"sharedMemoryMiB": 256,	PCIe接続時に使用するHugePage内確保サイズ[MegaByte]	確保するサイズは2のべき乗である必要あり。(基本"256"で良い)		
"imageURI": "localhost/cpu glue dma tcp:1.1.0",	使用するコンテナのコンテナイメージ名	CPUFunc用glue(DMA-TCP変換)処理モジュールのコンテナイメージ名。タグの数字はバージョン番号(基本変更不要)		
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	現状は2nd NIC利用を前提としているため値は"true"固定		
"virtualNetworkDeviceDriverType": "sriov",	ファンクションが2nd NICで利用する仮想NWデバイス用ドライバ	現状は"sriov"のみを前規としているため、設定値は"sriov"固定		
"template":{	作成するPodのテンプレートデータ	SUMMA SHOW SHOW BY ENTINEED CONSTRUCTION BY ALIEUS SHOW BOX.		
"apiVersion": "v1",	1F/00(9/9/FOGO) 270-1-7			
"kind": "Pod",				
"spec":{				
	D- 4マおチレナスコンニよに服士フラルウ持む			
"containers":[{	Podで起動するコンテナに関する設定情報	起動するコンテナは1台なので固定で良い		
"name": "cfunc-glue-fdma-to-tcp-1",				
"workingDir": "/opt/openkasugai-controller/sample-functions/functions-	コンテナのワーキングディレクトリ	値は"/opt/openkasuqai-controller/sample-functions/functions-ext/cpu qlue dma tcp"固定		
"command": ["sh","-c"],	使用するコンテナでの実行コマンド	実行コマンドの実態は"args"で指定している		
"args":["./build/glue",				
"%FORWARDING%",	コマンド実行時に渡す引数	CPUFunc用qlue(DMA→TCP変換)処理モジュールの実行コマンドとその引数		
"%WIDTH%",	3 (21 )(1)(1)(1)(2) (1)(2)			
"%HEIGHT%"],				
"securityContext":{				
"privileged": true		値は"true"固定		
},				
"lifecycle":{	lifecycleのフックに関する設定(ハンドラー)を定義する			
"preStop":{	コンテナが終了する直前のフック。コンテナ終了直前に実施する内容を定義する			
"exec":{	preStop時にコマンド実行を行うハンドラー			
"command": ["sh","-c", "kill -TERM \$(pidof ./build/qlue)"]}}},	Pod停止用コマンド	コマンドラインは「"sh","-c", "kill -TERM \$(pidof ./build/glue)"」で固定		
"volumeMounts":[{				
"name": "hugepage-1gi",	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolumeMount	值は"hugepage-1gi"固定		
"mountPath": "/dev/hugepages"		値は"/dev/hugepages"固定		
}.{		mare jumpepage mass		
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolumeMount			
"mountPath": "/var/run/dpdk"	7(7) jejori Cicigani, Cix/ii 9 abri bit Cix/ii 9 abri 10 71 7/ii 07 volumerioum	值は"var-run-dpdk"固定		
\$1.		REIO FOI TOIL OPON BINA		
"resources":{				
"requests":{				
requests :{				
"memory": "32Gi"		共有メモリ(hugepage)用。kBsの仕様「hugepageを使用する際には、CPUやメモリ(最低ど56か1つ)をリクエストしなければいけない」に対応するため設定。		
		値は任意で良いので"32Gi"固定で良い		
"limits":{				
"hugepages-1Gi": "1Gi"	hugepageの1枚分のページサイズ	値は"1Gi"固定		
}				
}				
}],				
"volumes":[{				
"name": "hugepage-1gi",	入力側のPCIe接続で使用する/dev/hugepagesディレクトリ用のVolume	上記volumeMounts."hugepage-1gi"と同じ値		
"hostPath":				
{"path": "/dev/hugepages"}		上記volumeMounts."hugepage-1gi"の"mountPath"と同じ値		
}.{		, <u>, , , , , , , , , , , , , , , , , , </u>		
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクトリ用のVolume	上記volumeMounts."dpdk"と同じ値		
"hostPath":	7.77 CICISAM CISCUS ADI DICCISCUS AT 1071 7/10 Volume	and the second s		
{"path": "/var/run/dpdk"}		上記volumeMounts."dpdk"の"mountPath"と同じ値		
\$1.		Engrotume-nounce, apar or mountrain Chaptile		
		現状は2nd NIC利用を前提としているため値は"false"で良い。		
"hostNetwork": false,				
		值は"true"固定		
"hostIPC": true,				
"hostIPC": true, "restartPolicy": "Always"		值は"Always"固定		
		值は"Always"固定		
		值は"Always"固定		

# DeviceInfoコントローラ用(赤太字箇所が環境に合わせて変更するポイント)

crc_deviceinfo_daemonset.yaml	説明	備考
apiVersion: apps/v1		
kind: DaemonSet		
metadata:		
name: crc-deviceinfo-daemon		
spec:		
selector:		
matchLabels:		
app: crc-deviceinfo-daemon		
template:		
metadata:		
labels:		
app: crc-deviceinfo-daemon		
spec:		
containers:		
- name: deviceinfo-container0		
image: localhost/deviceinfo:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_CLUSTERNAME	当該環境のk8sクラスタ名に関する環境変数	固定値
value: default		環境のクラスタ名に合わせて変更
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-deviceinfo-daemon		
volumes:		
- name: crc-deviceinfo-daemon		
hostPath:		
path: /etc/k8s_node/config		
type: File		

# PCIeConnectionコントローラ用 (赤太字箇所が環境に合わせて変更するポイント)

PCIeConnectionコントローラ用(赤太字箇所が環境に合わせ crc pcieconnection daemonset.yaml	説明	備考
apiVersion: apps/v1	<b>ब्रा</b> भा	VH
apiversion: apps/v1 kind: DaemonSet		
metadata:		
name: crc-pcieconnection-daemon		
spec:		
selector:		
matchLabels:		
app: crc-pcieconnection-daemon		
template:		
metadata:		
labels:		
app: crc-pcieconnection-daemon		
spec:		
containers:		
- name: pcieconnection-container0		
image: localhost/pcieconnection:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-pcieconnection-daemon		
- mountPath: /var/run/dpdk	PCIe接続の際に使用するDPDKで使用するディレクトリを設定	DPDKで使用するディレクトリを設定
name: var-run-dpdk		
- name: hugepage-1gi	hugepageとして用意したディレクトリの設定	想定環境では、HugePageとして使用するPCIe接続するので、その際の共有メモリ1枚分のページサイズを1GiBとして使っている。 当該workerノードにて設定している共有メモリ1枚分のページサイズによって、数字部分を変更する。
mountPath: /dev/hugepages		hugepageとして用意したディレクトリのファイルバスを設定
resources:		
limits:		
hugepages-1Gi: 16Gi		osで設定したhuqepaqeのサイズ。基本は左記の値のままで良い。
requests:		The state of the s
hugepages-1Gi: 16Gi		osで設定したhugepageのサイズ。基本は左記の値のままで良い。
memory: 1Gi		on carretorn adebades of 1548 metalograms may make a 250 g
volumes:		
volunies.		

# 4. CRCのdaemonset用YAML説明

- name: crc-pcieconnection-daemon		
hostPath:		
path: /etc/k8s_node/config		
type: File		
- name: var-run-dpdk	PCIe接続の際に使用するDPDKで使用するディレクトリを設定	
hostPath:		
path: /var/run/dpdk		DPDKで使用するディレクトリを設定
type: DirectoryOrCreate		
- name: hugepage-1gi	hugepageとして用意したディレクトリの設定	想定環境では、HugePageとして使用するPCIe接続するので、その際の共有メモリ1枚分のページサイズを1GiBとして使っている。 当該workerノードにて設定している共有メモリ1枚分のページサイズによって、数字部分を変更する。
hostPath:		
path: /dev/hugepages		hugepageとして用意したディレクトリのファイルパスを設定
type: DirectoryOrCreate		

# EthernetConnectionコントローラ用 (特に環境に合わせて変更する箇所は無い)

crc_ethernetconnection_daemonset.yaml	説明	備考
apiVersion: apps/v1		
kind: DaemonSet		
metadata:		
name: crc-ethernetconnection-daemon		
spec:		
selector:		
matchLabels:		
app: crc-ethernetconnection-daemon		
template:		
metadata:		
labels:		
app: crc-ethernetconnection-daemon		
spec:		
containers:		
- name: ethernetconnection-container0		
image: localhost/ethernetconnection:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-ethernetconnection-daemon		
volumes:		
- name: crc-ethernetconnection-daemon		
hostPath:		
path: /etc/k8s_node/config		
type: File		

# FPGAFunctionコントローラ用 (特に環境に合わせて変更する箇所は無い)

-PGAFunctionコントローフ用 (特に境境に合わせて変更する固所は無い)		
crc_fpgafunction_daemonset.yaml	説明	備考
apiVersion: apps/v1		
kind: DaemonSet		
metadata:		
name: crc-fpgafunction-daemon		
spec:		
selector:		
matchLabels:		
app: crc-fpgafunction-daemon		
template:		
metadata:		
labels:		
app: crc-fpgafunction-daemon		
spec:		
containers:		
- name: fpgafunction-container0		
image: localhost/fpgafunction:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-fpgafunction-daemon		
volumes:		

	- name: crc-fpgafunction-daemon			
	hostPath:			
	path: /etc/k8s_node/config			
	type: File			

GPUFunctionコントローラ用 (特に環境に合わせて変更する箇所は無い)

crc_gpufunction_daemonset.yaml	説明	備考
apiVersion: apps/v1		
kind: DaemonSet		
metadata:		
name: crc-gpufunction-daemon		
spec:		
selector:		
matchLabels:		
app: crc-gpufunction-daemon		
template:		
metadata:		
labels:		
app: crc-gpufunction-daemon		
spec:		
containers:		
- name: gpufunction-container0		
image: localhost/gpufunction:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-gpufunc-daemon		
volumes:		
- name: crc-gpufunc-daemon		
hostPath:		
path: /etc/k8s_node/config		
type: File		

# CPUFunctionコントローラ用 (特に環境に合わせて変更する箇所は無い)

CPUFunctionコンドローフ用 (特に環境に合わせて変更する固州 crc_cpufunction_daemonset.yaml	説明	備考
apiVersion: apps/v1		
kind: DaemonSet		
metadata:		
name: crc-cpufunction-daemon		
spec:		
selector:		
matchLabels:		
app: crc-cpufunction-daemon		
template:		
metadata:		
labels:		
app: crc-cpufunction-daemon		
spec:		
containers:		
- name: cpufunction-container0		
image: localhost/cpufunction:1.1.0		
imagePullPolicy: IfNotPresent		
securityContext:		
privileged: true		
args:		
- "kubeconfig=/kube/config"		
env:		
- name: K8S_NODENAME		
valueFrom:		
fieldRef:		
fieldPath: spec.nodeName		
volumeMounts:		
- mountPath: /kube/config		
name: crc-cpufunc-daemon		
volumes:		
- name: crc-cpufunc-daemon		
hostPath:		
path: /etc/k8s_node/config		
type: File		