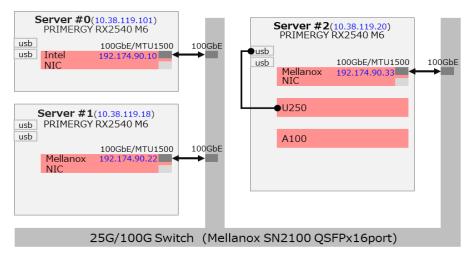
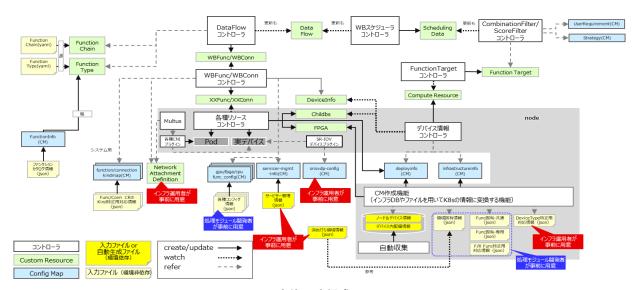
OpenKasugai-Controller Install Manual(Attachment)

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 VANU EKU	設定内容についての解説
2.YAML説明	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

v1.0.0



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 DataFlow YAML 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-fiter-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-P-VACM
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-XSSX-ア-7) CPUデコードPUCKOの問題子 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必得をない。 予定と、例:"からまれてもたけ、このでいきてニークな文字列を設定 予定と、例:"からまれてもたけった」でからまれてもたけった。「例:"からまれてもたけった。」では、「おまれてもてわらった。」では、「おまれてもている。「例:「からまれてもている」では、「例:「からまれてもている」では、「例:「からまれてもている」では、「例:「からまれてもている」では、「例:「からまれてもている」では、「例:「からまれている」では、「例:「からまれている」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている」では、「からまれている。」では、「からまれている」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれていったったったった。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、「からまれている。」では、
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の運用子。Functionsのmapのkev線を設定	TCP/IDPのボート番号ではない
connectionTypeName: "auto" - from:	データ送偏死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お送場戻で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:	テータ法領デFunctionの無別子、Functionsのmapのkey値を設定 データ法領デFunctionの出力ボート臨別番号(Functionが1出力の場合は0を指定) Connectionにおける法領先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/mink データ法債子(Functionの識別・Functions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がスクエッソー人を参加することはない
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 putchen ト 物図画像 目標 putchen の対 対 から可能 はな 育立 Connection L おする所像 アルロイロの作用 データルを展示putchenが用象 データルを展示putchenが用象がある。 データルを展示putchenが用象がある。 Connection L おする所 putchen put	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が対した場合は改善設立 Connectionと対して基準を表現をデルロに対しては、 データ機能予加には回路デールの開発を データ機能予加には回路デールの開発を データ機能予加には回る人がボール機能操作が正したが表現を データ機能予加には回る人がボール機能操作が正したが表現を Connection Typeのデースを記述する。	TCP/UDPO#-ト海号ではない TCP/UDPO#-ト海号ではない TCP/UDPO#-ト海号ではない 研究の配件が認知: Aud を指定。そのためConnectionTypeが2.92//ソースを参照することはない TCP/UDPO#-ト海号ではない
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-機能設定 データ機能子unctionの(選挙子 F. Austrone/map@ke-Map@ke- を発展子unctionの(選挙子 F. Austrone/map@ke-Map@ke- アータ機能子unctionの(基準子 F. Austrone/map@ke-Map@ke- アータ機能子unctionの(基準子 F. Austrone/map@ke-Map@ke- アータ機能子unctionの(基準子 F. Austrone/map@ke-Map@	TCP/UDPOM-ト海号ではない TCP/UDPOM-ト海号ではない 場合の場合が立流で、他ができ返す。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形プリ州田)が場合は、「who-end-of-chaint Toligita な文字外を設定すること 「少株的インド外界・人が下できかまた」で、2が実施を影合的で、いか・end-of-chaint Toligita な文字外を設定すること 「少株的インド外界・人が下できかまた」で、2が実施を影合は、「who-end-of-chaint Toligita な文字外を設定すること 「大林村・大手列子・人が下できかまた」で、2が実施を影合は、「who-end-of-chaint Toligita な文字列を記さすること
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可能放定 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表す。 データ機能デルロイビのの人がボート機能差しているのでは、 Connection Tay 17 エース または ** 10 本で 18 大田 18 世紀	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、自然に対象を開催を データ機能デルロの記述が、自然に関係を データ機能デルロの記述が、一般開催等を データ機能デルロの記述が、一般開催等を データ機能デルロの記述が、一般開催等を データ機能デルロの記述が、一般開催等を データ機能デルロの記述が、一般開催等を データ機能デルロの記述が、一般開催等を 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述が、一般開催を 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述が、一般開催を 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述が、一般開催を 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述が、一般開催を でのできないできない。 データ機能デルロの記述が、一般開催を でのできないできないできないである。 でのできないできないできないできないできないできないできないできないできないできない	TCP/UDPのボート番号ではない TCP/UDPのボート番号ではない 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*プログラスを受け、プログラスを受けるアンリ担当の場合は、"vib-end-of-chain"から独立をアデ列を設定すること 少多がようと対象とればた「マルビス」が「必要なが場合に、vib-end-of-chain"の地区を受けない。COCCが、アニークな文学列を設定する。 当度な、"vib-end-of-chain"を実には「大山内観音の場合を含む。vib-end-of-chain"などでから一切もでもたねは「アリア」 国際など、Wo-end-of-chain"を実に行っていることを受けませませます。 現代の場合に対象に、"auto"を確定、そのためConnectionTypeth2/04/97~スを参照することとない。 TCP/UDPのボート番号ではない 「CONNECTIONTOF/COSF*プログラスをCONNECTIONTOF/COSF*プログラスを受けませます。 TCP/UDPのボート番号ではない 「CONNECTIONTOF/COSF*プログラスをCONNECTIONTOF/COSF*プログラスを受けませます。 TCP/UDPのボート番号ではない 「CONNECTIONTOF/COSF*プログラスをCONNECTIONTOF/COSF*プログラスを受けませます。 TCP/UDPのボート番号ではない TCP
functionKey: "ufer-2" port: 0 connectionTypeRunne: "auto" connectionTypeRunne: "auto" functionKey: "ufer-1" port: 0 to: functionKey: "wb-end-of-chain-1" port: 0 connectionTypeRunne: "auto" - from: port: 0 connectionTypeRunne: "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。 データ機能デルロののに対した。 データ機能であるとのにあるとのにあるとのにあるとのにあるとのにあるとのにあるとのにあるとのに	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TOPでのボアブ((回際延興を受賞するがボアア)担当)の南向は、"wb-end-of-chain" から起きる文学外を設定すること - ************************************
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・有機器自作 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。 データ機能デルロののに対した。 データ機能であるとのにあるとのにあるとのにあるとのにあるとのにあるとのにあるとのにあるとのに	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TOPでのボアブ((回際延興を受賞するがボアア)担当)の南向は、"wb-end-of-chain" から起きる文学外を設定すること - ************************************
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 データ機能子がLandsの対抗・特別器を目代のLandsの対抗の関係 という機能を対している。 データ機能子がLandsの対抗・有効性を対している。 という機能子がLandsの対抗・力が開発 という機能子がLandsonemapticeの対抗力が開発 データ機能子のLandsonemapticeの対抗力が開発 データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力を開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の関係が データ機能子のLandsonemapticeの対抗力が開始の関係が データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を表現して データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子を表現したが データ機能子を表現した。 データ機能子のLandsonemapticeの データを表現を データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データを データ機能子のLandsonemapticeの データを デ	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デールではToneProsp (Rese を設定 データ機能デルロの対比が、一般調整 目光 United の対 はから単位できませ、 Connection、おける機能デルロのが開催 データ機能デルロの対能が、「本態を関係を関係を関係を関係を関係を関係を関係を Connection、おける機能デルロのでは、 Connection に対しる機能デルロのでは、 データ機能デルロのでは、一般では、 データ機能デルロのでは、データを データ機能デルロのでは、データを データ機能デルロのでは、 データを デ	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 データ機能子がLandsの対抗・特別器を目代のLandsの対抗の関係 という機能を対している。 データ機能子がLandsの対抗・有効性を対している。 という機能子がLandsの対抗・力が開発 という機能子がLandsonemapticeの対抗力が開発 データ機能子のLandsonemapticeの対抗力が開発 データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力を開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の関係が データ機能子のLandsonemapticeの対抗力が開始の関係が データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの対抗力が開始の発展で データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響 データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を開発して データ機能子のLandsonemapticeの影響を表現して データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データ機能子を表現したが データ機能子を表現した。 データ機能子のLandsonemapticeの データを表現を データ機能子のLandsonemapticeの データ機能子のLandsonemapticeの データを データ機能子のLandsonemapticeの データを デ	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, "urfer-2" port: 0 connectionTypeBalanne: "auto" connectionTypeBalanne: "auto" function(key: "urfer-1" port: 0 function(key: "whe-end-of-chain-1" port: 0 connectionTypeBalanne: "auto" - Brott: Dort: 0 to: function(key: "whe-end-of-chain-2" - Brott: 0 to: function(key: "whe-end-of-chain-2" - Brott: 0 connectionTypeBalanne: "auto" - Brott: 0 connect	データ機能デルロのの問題デェ AnationalmapReemitikgを データ機能デルロの対比が、中級開発 First ConditionalmapReemitikgを 「アータ機能デルロの対比が、中級開発 First ConditionalmapReemitikgを データ機能デルロの記憶デェルロのの記憶ディルので 第一プルルのでは、アータルの	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, "ufer-2" port: 0 connectionTypeRiame: "auto" SunctionKey; "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-inuctionType YAML self-inuctionType year self-inuctionType-decode self-in	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の開発・AutorionのImageReeの情報を データ機能力がLinkTonの開発・AutorionのImageReeの情報記 データ機能力がLinkTonの開発・AutorionのImageReeの情報記 データ機能力がLinkTonの開発・AutorionのImageReeの情報記 データ機能力がLinkTonの開発・Autorion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の情報と データ機能力がLinkTonのImageReeの情報と データ機能力がLinkTonのImageReeの情報記 データ機能力がLinkTonのImageReeの情報記 データ機能力がLinkTonのImageReeの情報記 データ機能力がLinkTonのImageReeの情報記 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の情報を データ機能力がLinkTonのImageReeの情報 フーザ件能能設定する ユーザ件能能設定する ユーザ件能能設定する ユーザ件能能設定する ユーザ件能能設定する ユーザがLinkTonoImageReeのIm	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, "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問題デェ AnationalmapReemitikgを データ機能デルロの対比が、中級開発 First ConditionalmapReemitikgを 「アータ機能デルロの対比が、中級開発 First ConditionalmapReemitikgを データ機能デルロの記憶デェルロのの記憶ディルので 第一プルルのでは、アータルの	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: "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 livint: "functionType yex-yex-yex-yex-yex-yex-yex-yex-yex-yex-	データ機能デルスの自動等デ、FunctionalmapReampleampleampleampleampleampleampleampl	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, "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」
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 このできないでは、対しては、Marchand では、Marchand で	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, "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 functionType YAML popt-selection transport port: 0 functionType YAML popt-selection transport port: 0 functionType (auto-functionType functionType functi	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、 「Connection」と対しては、 「Con	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」
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 to the second to the sec	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 このできないでは、対しては、Marchand では、Marchand で	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, "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、 「Connection」と対しては、 「Con	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、 「TCP/UDPのボー海岬ではは、 「TCP/UDPのボー海岬ではは、 「TCP/UDPのボー海岬ではは、 「COnnectionのTom/FCのは下点(機能操業を含するのボアリ型曲)の側点は、"wh-end-of-chain"から終える文字が手能でする。 「中央・サール・デール・デール・デール・デール・デール・デール・デール・デール・デール・デ

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:	10700000	
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・サイナのエリスを undicated for Confidence Operators Name を記す	
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 Europing Chair	
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/	
functionName: high-wifer functionIndCNRRRI remer: functionExplainer remer: functionExplainer remer: functionExplainer remer: functionExplainer remer: functionExplainer remeries remeri	Function/Landiffunction/Lanes 設定される機 Function/Lanes アンシン・グラ 着見れているFunction/Lanes (日本) Function/Lanes (アンシン・グラ 着見れているFunction/Lanes (日本) Function/Lanes (アンシン・グラ 着見れているFunction/Lanes (日本) Function/Lanes (日本) Lanes (日本	
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/	
functionName: high-enfer functionName: high-enfer ramer function legit-infer ramer gracino legit-infer functionName: legit-infer functionName: low-infer functionNam	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-tail or surption: 1.0 d	Function/hamiffunction/hamis 設定される様 Function/hamiffunction/hamis 設定される。 Function/hamis アンテンテンデルを表れている Function (info: Confide as) (function for American	通考
function/lame: high-enfer function/linCollegit: name: function/linCollegit: name: function-lample region regions - 1.0.0	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-leigh-wifer Inner	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
Enricionalme: high-rofe Enrici	Function/SandfunctionName 設定される機 FunctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Nameを報定 FunctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionNamespace(プァンプンシングを見まれているAutocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを対象を表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンク	備考
EurotsonName: high-wifer FunctionName: high-wifer InuntonindsCWRRI: remer: FaccionSwiph-wifer remer: TaccionSwiph-wifer repolversion: countingle.com/v1 repolversion: example.com/v1 repolversion: example.com/v1 repolversion: example.com/v1 repolversion: example.com/v1 repolversion: example.com/v1 repolversion: example.com/v1 remer: function-limpto-leve-wifer function-limpto-leve-wifer function-limpto-WRRI: name: function-limpto-leve-wifer name: function-decode	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(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Nameを報定 FunctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 DarctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを報定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionName(プァンプンシンが変更れているAutocontent/ConfineNap)がmediatal Namespaceを確定 RuctionNamespace(プァンプンシングを見まれているAutocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを対象を表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクのスタールを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 PunctionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクを表現し、Autocontent/ConfineNap)がmediatal Namespaceを確定 DartionNamespace(プァンプンシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンクシンク	通考
functionName: high-wifer functionName: high-wifer ramer functionName: high-wifer ramer function-high-wifer ramer function-high-wifer ramer function-high-wifer ramer function-high-wifer ramer function-high-wifer ramer functionName: how-wifer ramersace: widther-imagnee gee: functionName: low-wifer functionAmersace: widther-imagnee gee: functionName: low-wifer ramersace: widther-imagnee gee: functionName: low-wifer ramersace: widther-imagnee version: 1.0.0 widther-imagnee version: 1.0.0 widther-imagnee ramersace: widther-imagnee ramersace: widther-imagnee ramersace: widther-imagnee ramersace: widther-imagnee desire: ramersace: widther-imagnee desire: ramersace: widther-imagnee desire: ramersace: widther-imagnee data: ramersace: widther-imagnee	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 function-decode namespace: wohun-improc Editati dolph-role-decode namespace: wohun-improc Editati:	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	魔考
EurotsonName: high-wifer FunctionIndCNRRR: remer faccing-high-wifer ramespace: wifurc-imagine goe: functionName: low-wifer functionIndCNRRR: name: function-low-wifer ramespace: wifurc-imagine remer function-low-wifer ramespace: wifurc-imagine remer function-low-wifer ramespace: wifurc-imagine remer function-low-wifer remer function-wifer remer function-	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 function function in the second of the	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 対象が表す。 Function/hams/hams/hams/hams/hams/hams/hams/hams	
EncisionName: high-wife FunctionIndCNReit Transmer Tr	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: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer remestation: remember remestation: remember remestation: remember remestation: remember remestation: remember remestation: remember remestations-wifer r	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 Turcino-divorStreet: Transe: Surcino-decode Transe: Turcino-decode Turcino	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: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer reme: faccinch-sigh-wifer remestation: remember remestation: remember remestation: remember remestation: remember remestation: remember remestation: remember remestations-wifer r	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: Traction-high-wife reme: Traction-high-wife rames. Indirect-wingroc goe: function/indCNRRR rames (function-low-wifer rames. function-low-wifer reme: _apitersion-v1 kmic_configNap restables rame: function-becode ramespace: wbfunc-improc data: desipowaletemes: [{	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 FunctionIndexSides! Increase Tourish-winder Increase Tourish-windex Increase Tourish-winder Increase Tourish-windex Increase Tourish-winder Increas	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! In recommendation with winder remer function begin-winer remer function begin-winer remer function begin-winer versions 1.0.0 deforce improc versions 1.0.0 deforce improc contestables remer function begin	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 Tu	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 a service se	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	
EurotsonName: high-winer FunctionIndCNRRRI FunctionIndCNRRI FunctionIndCNR	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 Landing Landin High-role Landing Landin High-role	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 を使けるといのである。 Autoflookame/Tyn-2w-2/2ff を使けるといいのである。 Autoflookame/Tyn-2w-2/2ff を使けるといいのである。 Autoflookame/Tyn-2w-2/2ff を使けるといいのである。 Autoflookame/Tyn-2w-2/2ff を使けるといいのである。 Autoflookame/Tyn-2w-2w-2w-2w-2w-2w-2w-2w-2w	

"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>とouputInterfaceType>を使用する場合に、デ</majontypeをでは、couputinterfacetype></majontype></majontype></majontype>	
"configName": "fpgafunc-config-filter-resize-high-infer",	上記のAregionType>に配摘して上記のAmputInterfaceType>と <upre>このUputInterfaceType>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>に配備して上記のAinputInterfaceType>と<utputinterfacetype>とを使用する場合のファ</utputinterfacetype></upre>	
"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>	
)	ファ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>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>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 プロイに必要な場象の名前 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情報の名前 ファングションのスペック情報</outputinterfacetype>	現在心試作では未使用
"configName": "figafune-config-filter-resize-low-infer", "spectName": "spect" } spec: " "manuar": "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, "manCone": 1, "manCone": 8, "manCone": 40, "manCone": 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 プロイに必要である。 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 クランコンのスペック情報 ファンクラコンのスペック情報 ジアとファンラコンのスペック情報 ジアとファンラコンのスペック情報 がTriey-valueDysondプジェクトを要素とする影響をア学列にした様 中心に影響の養養を思考するための名前 使用するリソースの影響 衛用するリソースの影響 衛用するリソースの影響 電子の個大学展示を関するを必要して記述を表する 基本の個大学展示を「(for)	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": 1, "masCopstNum": 1, "navespectNum": 1, "n	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: " {	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: " "aneme": "spec1", "maCozer": 1, "maCozer": 1, "maCozer": 1, "maCozer": 40, "maCrapach/fisser": 40, "maCozer/baser": 40, "maCozer/b	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外にした機 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gonTypeに比較して上記のComputInterfaceTypeとそのUputInterfaceTypeとを使用する場合に、デ プロイに必要ではPopusに比較して上記のComputInterfaceTypeとそのUputInterfaceTypeとを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 選びため、プログラングの表示を 提供できないでは、Popus では、Popus では、Pop	現在の試合では未使用
"configName": "figafune-config-filter-resize-low-infer", "specName": "spec1" "spec": "anme": "spec1", "maCore": 1, "maSopart-for-specifiler-resize-log-infer-r	上記の-regionTypesに比較して上記の-inqutinterfaceTypesとを使用する場合に、デ プロイにの確認的場合語 上記の-regionTypesに比較して上記の-inqutinterfaceTypesとそのUputInterfaceTypes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 ジャングションのスペップ情報の話 対して、アングションの表では、アングションのスペック情報 を記述が必要を受けませます。 では、アングションの表では、アングションの表では、アングションの表では、アングションの表では、アングションの表では、アングションの表えが、アングションの表えたが、アングションの表表を表表が、アングションの表表を表表が、アングションの表表を表表が、アングションの表表を表表が、アングションの表表を表表が、アングランの表表を表表が、アングランの表表を表表が、アングランの表表を表表が、アングランの表表を表表を表表が、アングランの表表を表表を表表が、アングランの表表を表表を表表を表表を表表を表表を表表を表表を表表を表表を表表を表表を表表を	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-inquitinefaceTypesとのJuputinefaceTypesを使用する場合に、デ プロイにの確認的場合を開 上記の-regionTypesに比較して上記の-inquitinefaceTypesとくのJuputinefaceTypesを使用する場合のファ プランクションのスペック情報の ボール・アンクションのスペック情報 ジャングションのスペック情報 ジャングションのスペック情報 選問するリントンの最大者 選問するリントンの最大者 選問するリントンの最大者 選用を必要が必要が必要を使用するための名前 選邦を必要が必要が入力 ファンクションの最大人力展 ファンクションの最大人力展 ファンクションの最大人力展 ファンクションの最大人力展 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大人力度 ファンクションの最大力力度 上記の-regionTypesに定能して基金に使用可能を以力のインターフェース機関 上記の-regionTypesに定能して認めて使用可能を以力のインターフェース機関 上記の-regionTypesに定能して記念で使用可能を以力のインターフェース機関 上記の-regionTypesに定能して上記のより表現のませいからな、デ プロイに必要では同じたとに変化して記念で使用可能をは力のインターフェース機関 上記の-regionTypesとに関して上記のより表現のよります。	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 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。 「アングションの最大性力数 は下にかっていません。 「クリースケック情報 はアングションの最大性力数 は下にかっていません。 「クリースケック情報 はアングションの最大性力数 は下にかっていません。 「クリースケック情報 はアングションの最大性力数 はアングションの表していません。 はアングションのよりに対していません。 はアングランのよりに対していません。 はアングログランのよりに対していません。 はアングランのよりに対していません。 はアングランのよりに対していません。 はアングランのよりに対していません。 はアングランのよりに対していません。 はアングランのよりに対していません。 はアングランのよりに対していませんのようないのようないのよりに対していませんでいません。 はアングランのよりに対していませんのよりに対していませんのようないのよりに対していませんのようないのよりに対していませんのようないのようないのよりに対していませんのようないのようないのよりに対していませんのようないのよりに対していませんのようないのようないのよりに対していませんのようないのようないのようないのようないのよりに対していませんのようないのよりに対しませんのようないのようないのようないのようないのようないのようないのようないのようない	現在の試作では未使用
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1" } Spec: " "anne": "spec1", "macConfig-filter-resize-low-infer", "macConfig-filter-resize-low-infer", "macConfig-filter-resize-low-infer", "macConfigName": 10, "macConfigName": 10, "macConfigName": 11, "macConfigName": 11, "macConfigName": 11, "macConfigName": 11, "macConfigName": 11, "macConfigName": 12, "specName": "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uput/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 スポーツ・アンのスペップ機能の名割 スポーツ・アンのファンの機能の名割 スポーツ・アンのファンの機能の名割 スポーツ・アンのファンの機能の影響 を関するロンのの場合 を関するロンのの場合 関連するロンスの必要 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 関連するロンスのの場合 に対しているのでは、ロンスのの場合 ロンスのの場合の対象 コンスのの場合の対象 コンスのの場合 コンスのもののは、ロンスのの場合 コンスのもののは、ロンスのの場合 コンスのもののは、ロンスののは、ロンスのは、ロン	
"configName": "fignature-config-filter-resize-low-infer", "specName": "spec1" "spec": "spec1", "maCoze": 1,	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 プロイにの確認的場合の音 上記の-regionTypesに関連して上記の-input/interfaceTypesとそのUput/interfaceTypesを使用する場合のファ プランクランのスペック情報 プアンクランのスペック情報 プアンクランのスペック情報 プアンクランのスペック情報 プアンクランのスペック情報 プアンクランの表の大学 報酬するリンスの最か機能 報酬するリンスの最か機能 基本の数大地理像力(Pob) プアンクランの最大出力度 プアンクランの最大出力度 プアンクランの最大出力度 プアンクランの最大出力度 プアンクランの最大出力度 プアンクランの最大地では、アクトを要素とする配列を文字列にした機 deployabletemicをTypeを受けるための名前 概要の表別が表別の表別を受けるための名前 を期間では関連して、関連して場合に使用可能を认为のインターフェース機能 上記の-regionTypesに定慮して表記に使用可能を认为のインターフェース機能 上記の-regionTypesに定慮して表記した時間を以前のインターフェース機能 上記の-regionTypesに定慮して表記した時間を以前のインターフェース機能 上記の-regionTypesに定慮して表記の表に使用可能を认为のインターフェース機能 上記の-regionTypesに定慮して表記のよりのよりからいとない。 上記の-regionTypesに定慮して表記のよりのよりからない。 上記の-regionTypesに定慮して表記のよりが表記を表記を表記を表記を表記を表記を表記を表記を表記を表記を表記を表記を表記を表	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 以下は中ではないのでは、このでは、このでは、このでは、このでは、このでは、このでは、このでは、こ	
"configName": "fignature-config-filter-resize-low-infer", "spectName": "spect" "spect" "anne": "spect", "macCore": 1, "macCore 1, "spect" "spect" "spect" "spect" "spect" "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": "figalune-config-filter-resize-low-infer", "spect Name": "spect " " spect " " "manufer": "spect ", "maxCoard-st 1, "maxCoard-st 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 10, "maxCapachyBase": 10, "maxInputhor": 1 "maxCapachyBase": 10, "maxCapachyBase": 11,	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 Spont 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 Spont プラェクトを要素とする配列を文字列にした機 の中医型が必要素を開するための各種 使用するリソースの最大機 基本の最大地 大学のでは、「クランカンの人を受け、 「クランカンの人を受け、 「クランカンの人の人の人を受け、 「クランカンの人の人の人を受け、 「クランカンの人の人の人の人の人の人の人の人の人の人の人の人の人の人の人の人の人の人の	
"configName": "figadune-config-filter-resize-low-infer", "specName": "spec1" "spec", "manufacture-til, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 8, "maxCapachyBase": 9, "maxCapachyBase": 9, "maxCapachyBase": 9, "maxCapachyBase": 1, "special "capachyBase": 1, "special "capachyBase": 1, "special "capachyBase": 1, "maxCapachyBase": 1, "maxCapachBase": 1, "maxCapachBase"	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とそ 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の音が一般 健用するリンテムの最小機 基本の最大地理像力(Pob) ファンクタンタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力を ファンクタンの最大力を ファンクタンの最大力を ファンクタンのないの情報に使用である。 に見かって映画が下がでは、日本のでは、中では、中では、中では、アラスを使用する場合に、デ ファンクタンの場合と関係して最后に使用である。カウスティス機能 上記の-regionTypesに定能した場合に使用であるカウンターフェース機能 上記の-regionTypesに定能した場合に使用である。カウスティス機能 上記の-regionTypesに定能した場合に使用であるカウンターフェース機能 上記の-regionTypesに定能してまた。 エ記の-regionTypesに定能してまた。 エ記の-regionTypesに定能してまた。 ファンタンタンのスペック情報 以下をアッカルログラムでは、アラスを使用する場合のファ ファンタンタンのスペック情報 以下をアッカルログラムでは、アラスを使用する場合のファ ファンタンタンのスペック情報 以下をアッカルログラムである。 ファンタンタンのスペック情報 以下をアッカルログラムである。 関本のサースの最大機 関本のサースの最大機 基本の最大効態をプラスのある。 基本の最大効態をプラスクを発	
"configName": "figadune-config-filter-resize-low-infer", "specName": "spec1" } spec: " "mare": "spec1", "marCore": 1, "marCore": 1, "marCore": 1, "marCore": 1, "marCore of the core of th	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とそ 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の音が一般 健用するリンテムの最小機 基本の最大地理像力(Pob) ファンクタンタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力度 ファンクタンの最大力を ファンクタンの最大力を ファンクタンの最大力を ファンクタンのないの情報に使用である。 に見かって映画が下がでは、日本のでは、中では、中では、中では、アラスを使用する場合に、デ ファンクタンの場合と関係して最后に使用である。カウスティス機能 上記の-regionTypesに定能した場合に使用であるカウンターフェース機能 上記の-regionTypesに定能した場合に使用である。カウスティス機能 上記の-regionTypesに定能した場合に使用であるカウンターフェース機能 上記の-regionTypesに定能してまた。 エ記の-regionTypesに定能してまた。 エ記の-regionTypesに定能してまた。 ファンタンタンのスペック情報 以下をアッカルログラムでは、アラスを使用する場合のファ ファンタンタンのスペック情報 以下をアッカルログラムでは、アラスを使用する場合のファ ファンタンタンのスペック情報 以下をアッカルログラムである。 ファンタンタンのスペック情報 以下をアッカルログラムである。 関本のサースの最大機 関本のサースの最大機 基本の最大効態をプラスのある。 基本の最大効態をプラスクを発	
"configName": "fignature-config-filter-resize-low-infer", "spectName": "spect", "spectName": "spect", "farme": "spect", "macCare": 1, "macCare": 1, "macCare": 1, "macCare": 1, "macCare filter-resize-logh-infer "macCare filter-resize-logh-infer "macCare filter-resize-logh-infer "macCare filter-resize-logh-infer "macCare filter-resize-logh-infer namespece: wofunc-insproc data: deployabiliteria: "["farme": "spect", "reportifyer": "part 100 godyeter", "outputtine-filter-type: "hest 100 godyeter", "configName": "cputunc-config-filter-resize-logh-infer", "spect filter-resize-logh-infer", "configName": "cputunc-config-filter-resize-logh-infer", "spect filter-resize-logh-infer", "macCare": 1, "macCar	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を認めらお問 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を提出している場合を表現しているのでは、 に対しているのでは、 に対してい	
"configName": "figadune-config-filter-resize-low-infer", "spect Name": "spect " "spect" "anie": "spect", "maCoard-stand-low-filter-resize-low-infer", "maCoard-stand-low-filter": 8, "maxCapath-filter-resize-low-infer", "specth-filter-resize-low-infer", "specth-filter-resize-low-infer", "specth-filter-resize-low-infer", "maxCapath-filter-resize-low-infer	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 は本の最大が観光を変更を繋ぎるための名様 現まるリンースの最小値 使用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小値 の用するリンースの最小性 は、本の最大が観光を変更を繋ぎるための名様 本本の最大が観光を表示的機能) アンングションの最大人力数 ファングションの最大人力数 ファングションの最大人力数 ファングションの最大人力数 ファングションの最大人力数 ファングションの最大人力数	
"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 を表示の最大が要素で作り、 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 pput を表示の最大が表現を受けることの名様 を表示の最大が関係が作り、 ファンクションの最大が表現を受けませませます。 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 daployabeltemaを対の要素を繋ぎるための名様 に最初では対象があります。 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 daployabeltemaを対の要素を受けまるための名様 に最初では最初を見からに関心して上部の一体の子のプラーフェース機関 上記の-regionTypesに定断して上部の上部のインターフェース機関 上記の-regionTypesに定断して上部の上部のでクターフェース機関 上記の-regionTypesに定断して上部の上部のインターフェースを受けませます。 上記の-regionTypesに関心して上部の上部のよりに対象があるのようなスペック解析の名様 エとの子の子のより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示を	
"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(Pis) プアンクションの最大力を プアンクションの最大力を プアンクションの最大力を プアンクションの最大力を プアンクションの最大力を プアンクションの最大力を に関いては、1000年の大力を は下をヴァンタンを 「と思の-regionTypesに定能した場合に使用可能なよりカッイシャーフェース機関 上記の-regionTypesに定能した場合に使用可能なよりカッイシャーフェース機関 上記の-regionTypesに定能して上部の-regionTypesと定能して生活に使用で使用可能などのインターフェースを プアンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションのスペック開催 ファンクションの表大力が度 ファンクションの最大力が度 レースを呼吸が使用では、サービルでは、中では、サービルを使用する場合のファンクションの最大力が度 ファンクションの最大力が度 レースを呼吸が使用が表が表が多ります。とそのなら前に レースを使用するといのなどのよりに対している場合 ファンクションの最大力が度 レースを呼吸が表がませ、サービルを呼吸をはからないのカットとを使用するとのなが表が表が表が表が表が表が多ります。とを受けまたのなら前に関するとないのなどのようによりによるとしている。 レースを使用では、1000年の日本では、10	
"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は、日本のよりに対しませない。 はアルタンターのよりに関ルでは、日本のよりに対しませない。 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	
"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: " { "ranee": "spec1", "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": 8, "maxCoard-overlage": 9, "maxCoard-overlage": 9, "maxCoard-overlage": 9, "maxCoard-overlage": 1, "maxCoard-overlage": 1, "specifier-resize-log-inder": 1, "specifier-resize-log-inder": 1, "maxCoard-overlage": 1, "maxCoard-overlage: 1, "ma	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開催の名前 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンのスペック開催 リアングラョンの最大権 基本の最大者を同かの表面と構 基本の最大者を同かの表面と対象 メールを表示の表面を対象を関するための名前 クアングラョンの最大地方数 リアングラョンの最大地方数 リアングラョンの表示の関係と関係しておきないのようには、サング・アングラョンの最大地方数 リアングラョンのスペック情報の名前 リアングラョンのスペック情報の名前 リアングラョンのスペック情報の名前 リアングラョンのスペック情報の名前 リアングラョンのスペック情報の名前 リアングラョンのスペック情報の名前 リアングラョンの表示を使用する場合にデアングラョンの最大地方数 リアングラョンの最大地方数 リアングラョンの最大人方数 リアングラョンのまた。日本のまたのようのまた。アングラョンの最大人方数 リアングラョンの最大人方数 リアングラョンの表大人方数 リアングラョンの表大人が対象	
"configName": "figadum-config-filter-resize-low-infer", "specName": "spec1" } spec: "{ "rame": "spec1", "maxCoard-st., "maxCoard-st.,	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、 ファンクションのスペック解除 以下top-valuatringのでは、 のputitierface/type>を使用する場合のファ ファンクションのスペック解除 以下top-valuatringのでは、 のputitierface/type>を使用する場合のファ ファンクションのスペック解除 以下top-valuatringのでは、 のputitierface/type>を使用する場合のファ ファンクションのスペック解除 以下top-valuatringのでは、 のputitierface/type は本の最大が意味が作り、 ファンクションの最大地 大変 は本の最大が意味が作り、 ファンクションの最大が 大変 ファンクションの最大が は本の最大が意味が作り、 ファンクションの最大が は本の最大が意味が作り、 ファンクションの最大が はまり、 に関いていていていていていていていていていていていていていていていていていていて	現在の試合では未使用
"configName": "figadum-config-filter-resize-low-infer", "specName": "spec1" } spec: " { "ranee": "spec1", "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": 8, "maxCoard-overlage": 9, "maxCoard-overlage": 9, "maxCoard-overlage": 9, "maxCoard-overlage": 1, "maxCoard-overlage": 1, "specifier-resize-log-inder": 1, "specifier-resize-log-inder": 1, "maxCoard-overlage": 1, "maxCoard-overlage: 1, "ma	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で使用ではたがのインターフェース推断 上記のでsgonTypesに発見しては他に一般ので使用でありたシャンクリーのよりを使用する場合に、デ 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 使用するリンアンの意とないまった。 エドをいっないまった。 エドをいっないまった。 エドとのでsgonTypesに関連しては、自然では明可能とようのインターフェース推断 エドとのでsgonTypesに関連しては自然で使用可能とようのインターフェース推断 上記のでsgonTypesに関連しては自然で使用可能とようのインターフェースを開 コア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の名前 コアンクションのスペック情報の名前 コアンクションのスペック情報である前	現在の社会では未使用
"configName": "fignafune-config-filter-resize-low-infer", "speckiame": "speci" "speci": "speci": "mancare": "speci": "speciment vi "speci": "speci"	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-に定機して基礎とで開発に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して基礎との信息と呼吸をは対したインターフェース機関 プロイにの接ば機能の名前 上記の-responTrypes-に定機して基礎の-に向いた同様は対しません。デ	
{ "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推開 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推開 上記の-creptonTypeンに定慮した場合に使用可能な出力のインターフェース推開 上記の-creptonTypeンに定慮して接合に使用可能な出力のインターフェース推開 上記の-creptonTypeンに定慮して決定の-creptonTypeンと-couputInterfaceType>を使用する場合に、デ プロイに必要が開発します。	
{ "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>に設備した場合に使用可能な力のインターフェース専列 上記の-responType>に設備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>に設備して上記の-input interfaceType> そのはput interfaceType>を使用する場合に、デ フロイに必要が扱わるに対 上記の-responType>に記備して上記の-input interfaceType>と<のはput 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	
{ "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-に定機して基礎とで開発に使用可能な出力のインターフェース機関 上記の-responTrypes-に定機して基礎との信息と呼吸をは対したインターフェース機関 プロイにの接ば機能の名前 上記の-responTrypes-に定機して基礎の-に向いた同様は対しません。デ	
{ "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なはカナの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なはカナの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なはかの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/Fio/要高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な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に発展した場合に使用可能は入力のインターフェース模別 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模別 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模別 上記の-creptompeesに記憶した比較の-creptompeesに対してよりで表現しました。デ したの-creptompeesに記憶した記念-creptompeesに対したのであった。デ したの-creptompeesに記憶した記念-creptompeesとその中の記念は出力のインタンコンのスペップ情報の名前 上記の-creptompeesに記憶した場合に使用可能は入力のインターフェース模別 上記の-creptompeesに設備した場合に使用可能は入力のインターフェース模別	
{ "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。アフィースを呼吸を対力の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。アフィースを呼吸を対力の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。アフィースを呼吸を必要がある。アフィースを呼吸を表	
{ "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シターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカナウスターフェース機関 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 位置的なまでは機能が 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカウィンターフェース機関 上足の-responTypes-LC機能した影響に使用である。サースによりでは、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、日本	
{ "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", "regonType": "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シターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカナウスターフェース機関 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 位置的なまでは機能が 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカカウイシターフェース機関 上足の-responTypes-LC機能した場合に使用可能とはカウィンターフェース機関 上足の-responTypes-LC機能した影響に使用である。サースによりでは、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、日本	
{ "name": "Rem1", "regonType": "16", "publisher Exceptes" "host 100 gether", "outputitien faceType": "host 100 gether", "outputitien 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 を確認をは関係数 と思わってeponTropeとに登離した場合に使用可能な入力のインターフェース機関 と思わってeponTropeとに登離した場合に使用可能な入力のインターフェース機関 と思わってeponTropeとに登離したと響点に使用可能な対力のインターフェース機関 と思わってeponTropeとに登離してと思われます。 プロイにの影響が終める前 と思わってeponTropeと定離して上記の、creputInterfaceStype>と < couputInterfaceStype>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 を確認では関係的 と思わってeponTropeと定離して上記の、creputInterfaceStype>と < couputInterfaceStype>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 と認めてeponTropeと定離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、creputIntypeと定離した場合に使用可能なよ力がインターフェース機関 上記の、creputIntypeと定離した場合に使用可能なよ力がインターフェース機関 上記の、creputIntypeと定義した。これの、creputInterfaceStype>を使用する場合に、デンタイと、必要はManageStype とてののはManageStype と述るこれをいまった。 これの、creputIntype となるこれをいまった。 これの、creputIntype には、creputIntype となるこれをいまった。 これの、creputIntype には、creputIntype には、cre	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 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 なための場合が開発を発 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-に定機した場合に使用可能な対象がインターフェース機関 上記の-responTypes-に定機してよることが表現しませんでは対象が存在がType>を使用する場合にデ 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 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 コアンクミコンのスペック情報 コアンクミコンのスペック情報 コアンクミコンのスペック情報 コアンクミコンのスペックを表現した。日本に使用ではないませんではないませ	現在の試作では未使用
{ "name": "Rem1", "regonType": "Net 100gether", "noutultnerfaceType": "host 100gether", "outputtnerfaceType": "host 100gether", "configName": "gputunc-config-low-infer", "spec Name": "spec 1" "}, { "name": "Rem1", "regonType": "Net, "noutultnerfaceType": "mem", "outputtnerfaceType": "host 100gether", "configName": "gputunc-config-low-infer", "spec Name": "spec 1" "spec Name": "spec 1" "spec Name": "spec 1" "mande": "spec 1", "mande": "spec	deployableItemedsが少更高を参照するための名前 位置可定は回路間 上記の-responTypesとに整理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とに整理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とに整理した場合に使用可能は対力がクシーフェース推列 上記の-responTypesとに推出した基色の cinquatinterfacelTypesと couputInterfacelTypesを使用する場合に、デ プロイに最近が開始の名前 deployableItemedsが少更多を参照するための名前 位置的収益が開始がありか要素を参照するための名前 位置的では自然では短期でよりに関連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とに整理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とに整理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とに整理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とに整理した場合に使用可能とな力のインターフェース推列 上記の-responTypesとに整理した場合に使用可能と対力のインターフェース推列 上記の-responTypesとに整理した場合に使用可能と対力のインターフェース推列 上記の-responTypesとに整理した場合に使用可能と対している。 「上記の-responTypesとを提出した場合に対しているのでは、 ファンクションのスペック情報の名前 以下を呼ゃかぬかけらな多数のスペック情報 以下を呼ゃかぬかけらな多数のなります。	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 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 なための場合が開発を発 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-に定機した場合に使用可能な対象がインターフェース機関 上記の-responTypes-に定機してよることが表現しませんでは対象が存在がType>を使用する場合にデ 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 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 コアンクミコンのスペック情報 コアンクミコンのスペック情報 コアンクミコンのスペック情報 コアンクミコンのスペックを表現した。日本に使用ではないませんではないませ	現在が出る方式
{ "name": "Rem1", "regonType": "16", "regonType": "16", "noutDutterfaceType: "host100gether", "outputterfaceType: "host100gether", "corfgName": "goutunc-config-low-infer", "specName": "goutunc-config-low-infer", "specName": "spect" "name": "Rem1", "regonType: "1", "noutputterfaceType": "host100gether", "sopothier EnderType": "host100gether", "specName": "spect " "spec Name": "spect" "manCapeType": "spect", "manCapeType": "nameType: "nameType": "nameType: "noutputterfaceType": "nout	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動的という	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" "name": "spec 1" "name": "spec 1" "name": "spec 1", "name": "spec 1", "name Oats Thought Type 1", "name Oats Though Type 1", "	deployable/timeds/Prop事業を参照するための名前 を確認は近時機関 上記の-region/poes上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/poes上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/poes上に関連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-region/poes上に関連して上記のよりに対しませんできた。 アプロイに参考が動から新 上記の-region/poes上に関連して上記のようのはInterfaceStypesを使用する場合に、デ アプロイに参考が動から新 上記の-region/poes上に関連して上記のようの名前 を信仰では「知識を持つの表表を影響するための名前 を信仰では「知識を持つの表表を影響するための名前 と思うでregion/poes上に関連して上記のようのは「Tender Administration Adm	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動的という	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": "sputur-config-low-infer", "specName": "spec1" }, """ "name": "spec1" "name": "spec1" "negion*/pet": "mem", "output/itemface*/pet": "nem", "output/itemface*/pet": "nost100gether", "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/pos-LC機関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/pos-LC機関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/pos-LC機関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-region/pos-LC機関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-region/pos-LC機関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-region/pos-LC機関した影響と、region/pos-とくoutput/interfacePypos-を使用する場合に、デ ンクシュンのスペック開催の名前 を信仰をは「知識を持つを乗るを影響するための名前 を信仰では「知識を持つの要素を影響するための名前 を信仰では「知識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-region/pos-LC機関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-region/pos-LC機関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-region/pos-LC機関した影響に使品に使用可能な力力のインターフェース機関 上記の-region/pos-LC機関した影響に使品に使用では対力のインターフェース機関 上記の-region/pos-LC機関した影響に使品に使用では対力のインターフェース機関 上記の-region/pos-LC機関した影響に使品に使用では対力のインターフェース機関 上記の-region/pos-LC機関した影響に対して影響となります。 上記の-region/pos-LC機関した影響に使用では対力を含めらア ンクションのスペールの機能 単年本の最大の多点の影響を表するための条件 単年本の単大権の関本を対しているの表す 単年本の最大機能を対すいてはい意)。回路のチャネル機等によって許まる 基本の最大的を表すと表す。	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

対定する 定する 用するFunctionChainのmetadata.Name 用するFunctionChainのmetadata.Namespace
定する 用するFunctionChainのmetadata.Name
定する 用するFunctionChainのmetadata.Name
定する 用するFunctionChainのmetadata.Name
定する 用するFunctionChainのmetadata.Name
用するFunctionChainのmetadata.Name
n y or unction chall of netadata. Namespace
に満たす必要がある要件を記載する
ン全体の要件を記載
シを記載
b/で _百 口申以
nctionの識別子
CCIOFIの自成が引
B. JOS. J. MICONEZIN Z.L. TRICHIOZ
。Podの2nd NICのIPアドレスとして設定される
ィルタリサイズ)のIPアドレス
イルタリサイズ)のポート番号
ズFunctionの識別子
。Podの2nd NICのIPアドレスとして設定される
h岐) のIPアドレス
h岐) のポート番号
tionの識別子
。Podの2nd NICのIPアドレスとして設定される
。前段のファンクションとのTCP接続の確立のために必要
2
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
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
unction1の識別子
。Podの2nd NICのIPアドレスとして設定される
。GStreamerのビデオ処理のコマンド実行(fpgdepay)に必要
信ツール)のIPアドレス
信ツール)のポート番号
信ツール)のポート番号
信ツール)のポート番号
信ツール)のボート番号 unction2の識別子
信ツール)のボート番号 unction2の遺例子 。Podの2nd NICのIPアドレスとして設定される
信ツール)のボート番号 unction2の識別子 。 Podの2nd NICのIPアドレスとして設定される 。 GStreamerのビテオ処理のコマンド実行(fpgdepay)に必要 3
作。Podの2nd NICのIPアドレスとして設定される。 GStreamerのビデオ処理のコマンド実行(fpgdepay)に必要
信ツール)のボート番号 unction2の遺例子 。Podの2nd NICのIPアドレスとして設定される 。 GStreamerのビデオ処理のコマンド実行(fpgdepay)に必要 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
スス

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outputlPAddress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「8.7 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 自身のポート番号

② inputPort: 5004 ③ outputIPAddress: 192.174.90.111

送信先のIPアドレス ④ outputPort: 15000 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。

	-Man	環境に合わせた	開連 / Pop Pet 27 部 中心 Pop 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
predetermined-region.json	説明	変更が必要※	備考(どの値を記載すれば良いかなど)
			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	
			・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
"deviceUUID": "21330621T013",	その領域があるデバイスの識別情報	0	・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
			・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": "lane0",	その領域の領域名	0	・FPGAの場合: "lane"+"\${lane番号}"を記載する
Subperior Filanco /	Continuo titrates	Ü	- CPU/GPUの場合: deviceTypeと同じ値を記載する
			·FPGAの場合:以下のフォーマットで記載:「"\${デバイス種別}" + "-" + "\${親bs-id}" + "-" + "\${lane数}" + "lanes" + "-" + "\${NIC数}" + "nics"]
			・サンブルとして用意しているFPGA回路(OpenKasugai-Controller-InstallManualの0.4節に記載)を用いる場合は、
"regionType": "alveou250-0100001c-2lanes-0nics"	その領域の領域種別	0	デバイス種別="alveou250", 親bs-id="0100001c", lane数="2", NIC数="0"で固定なので、
			regionTypeは必ず"alveou250-0100001c-2lanes-0nics"となる。
			- CPU/GPUの場合: deviceTvpeと同じ値を記載する
},			
{	1領域分の情報		
"nodeName": "worker0",		0	
"deviceUUID": "21330621T013",		0	
"subDeviceSpecRef": "lane1",		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
},			
{	1領域分の情報		
"nodeName": "worker1",		0	
"deviceUUID": "21330621T00Y",		0	
"subDeviceSpecRef": "lane0",		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
<i>},</i>			
"nodeName": "worker1".	1領域分の情報		
"nodeName": "Worker1", "deviceUUID": "21330621T00Y".		0	
"subDeviceSpecRef": "lane1".		0	
		0	
"regionType": "alveou250-0100001c-2lanes-0nics"		0	
j,	1領域分の情報		
"nodeName": "worker1",	1 1月45次77 (27) 門平牧	0	
"deviceUUID": "qpu-702fb653-43a4-732d-6bc4-7b3487696c90"		0	
"subDeviceSpecRef": "a100",		0	
"regionType": "a100"		0	GPU/CPUの鎖域権別の値はデバイス種別(deviceType)と同等
regiontype . a100		0	Gro/Growingの他はアハコス連加(device)pe/に呼呼
1			
1	II	1	

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。てれ以外のナバイスを使用する場合は足能が必要。		
devicetypemap.json	説明	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"代"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はStream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-强体制大强。
}]	Control (or () - ATTINUESD	THE PROPERTY OF THE PROPERTY O
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) CoOn (x)
	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とつりではデバイス権別(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。偏は暫定権。
		= 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。

追加アハ1人力の 刺乳に対する情報の追記が必要。 追記を行う場合の追記力法は個名側を参照のこと。		
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"と"IX"に関して、达信元もいいは达信光との接続ダイノが侵敛考えられる		備考
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Jlaneにつき1Functionのみ filter/resizeの場合Jlaneにつき1Functionなで、この簡はJane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 inieutolytyを提供する応信用リケー人件	
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-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。

GPI Funtion 田堆論処理モジュール(享度堆論を実施)のコンフィグ情報

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 gpufunc-config-high-infer.json	説明	備考(どの値も基本変更不要)
gpurunc-conng-nign-inter.json		帰う (Cの他も本々文で小女)
"rxProtocol": "DMA",	入力側がDMA通信、出力側がRTP通信を行う場合の設定情報 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:1.0.0",	使用するコンテナのコンテナイメージ名	WBI IKP 3 ジプト 1 A 10 2 2 グービス・ビッグ・ログラング 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1
"additionalNetwork": true,	ファンクションの2nd NICの利用の有無	間及はmmを大売でするCI OTION Internation 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	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パス	Telly Cytemic 失力によっている。 Tell Type Telly Cytemic 失力によっている。 Tell Type Celly Telly Cytemic 大力によっている。 Tell Type Celly Telly Cytemic Cyte
"CUDA MPS LOG DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルパス。	
	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	
"SHMEM_SECONDARY": "1",	御するための情報。	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ダリモードは"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
},	7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	
"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",	11777	
"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-Volo && gst-launch-1.0 -ev ftpgasrc !", "Video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%", "! nvvideoconver! 'video/x-raw(memory:NVMM), format=(string)RGBA",		
"I m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1", "%WIDTH%", "%HEIGHT%",		
"! queue ! 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 !	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ブラグインの実行コマンドと引数
<pre>ivvideoconvert !",</pre>		
"%OUTPUTIPS", "OUTPUTIPS", "SYNC=True"],		
"securityContext":{		
"privileged": true		值は"true"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	
"mountPath": "/dev/hugepages"		值は"/dev/hugepages"固定
}.{		
"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
}.{		
"name": "dpdk",	入力側のPCIe接続で使用するDPDKで使用するディレクドリ用のVolumeMount	Thus I work
"mountPath": "/var/run/dpdk"		値は"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"326"固定で良い
},		
"limits":{		
"hugepages-1Gi": "1Gi"		值は"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
M. Harris Maria Maria Maria	MDCD MDCWWRBTHT FACTOR A DILLIDON ()	
"\" "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
name": "host-nvidia-mps", "hostPath": {"path": "/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Volume	上記volumeMounts."host-nvidia-mps"と同じ値 上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値

"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 J. G. C. G. C. G. C. G.	and the second of the second o
{"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,		
"restartPolicy": "Always"		
}		
}		
},		
{	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/gpu infer tcp:1.0.0",	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	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	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パス	
"CUDA MPS LOG DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルパス。	
"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",	コンナナドリCのGStredmerノフソインの行物アイレクトリ。	
"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/DepStream-Yolo && gst-launch-1.0 -ev fpgadepay", "%INPUTP%", "%INPUTPORT%", "! video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%", "! n.videoconvert ! video/x-raw(memory:NVMM), format=(string)RGBA", "! m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1", "%WIDTH96", "%HEIGHT%", "%HEIGHT%", "! queue ! 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", "%GOUTPUTPORT%", "\$NO_CTRUTP", "sno_ctrue"]. "securityContext":{	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プラヴインの実行コマンドと引数
"privileged": true		
"volumeMounts":[{	MDCB MDC機能開大相下で展行すフォルのディカリリBのバー	
"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	理控系数UCUDA MAC DIDE DIDECTODU/IUCUI 在
"mountPath": "/tmp/nvidia-mps"		環境変数"CUDA_MPS_PIPE_DIRECTORY"と同じ値
}]		
}],		
"volumes":[{	MDCEL MDC///WRITT-IPT-YE/C-+75-11-2" - 51-11-20-11	LETURING Manufacturidis and the Color
"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":		LETURING Manufacturidis and "Ollan unblaktiticili" (5
{"path": "/tmp/nvidia-mps"}		上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値
3]		
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。
"hostNetwork": false, "hostIPC": true,		現れはZhu NiCや用で制度としているため間は ldise で良い。
"hostNetwork": false,		現れはZNU NICや別州で削減ZCU しいのにめ間は Idise CERUs
"hostNetwork": false, "hostIPC": true,		現れはZNU NICや別州を削減をCU しいかため間は IdiSe CERい。
"hostNetwork": false, "hostIPC": true,		現れはZHU NILCHJHYで削減をCU Cいりにめ間は Idise CERUs

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/qpu infer dma:1.0.0",	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": "srioy",	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パス	
"CUDA_MPS_LOG_DIRECTORY": "/tmp/nvidia-mps", "SHMEM_SECONDARY": "1",	MPSのログ出カ用ディレクトリのフルパス。 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	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。
"HEIGHT": "416",	入力映像のフレームサイズ(高さ)。	推論がアジリキのアプリスティー 日本 リ で、FCLE技術のコンドローブと座所自主するビルアアデニーは 1 で改定する。 基本 1 で改定する。
"WIDTH": "416"	入力映像のの入力フレームサイズ(幅)。	************************************
}.	The state of the s	
"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",		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	値に"/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"], "args":["cd /opt/nvidia/deepstream/deepstream-7.0/sources/objectDetector_Yolo/ && gst-launch-1.0 -	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
ev fpgasrc!", "video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%", "I nvvideoconvert! 'video/x-raw(memory:NVMM), format=(string)RGBA'", "I m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1", "%WIDTH%", "%HEIGHT%", "queue! nvinfer config-file-path=./config_infer_primary_yoloV3_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", "%OUTPUTIP%",	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プラヴインの実行コマンドと引数
"%OUTPUTPORT%", "sync=true"], "securityContext":{		
"privileged": true		値は"true"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	値は"hugepage-1gi"固定
"mountPath": "/dev/hugepages"		値は"/dev/hugepages"固定
}.{	Woles In the second sec	
"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	環境変数"CUDA MPS PIPE DIRECTORY"と同じ値
"mountPath": "/tmp/nvidia-mps"		環境妥数"CUDA_MPS_PIPE_DIRECTORY"で同い個
},{ "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"	入力側のPCIe接続で使用するDPDKで使用するテインケーク用のVolumeMount	値は"var-run-dpdk"固定
}1.		mare the three particulars.
"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
}, "limits":{		
"limits": { "hugepages-1Gi": "1Gi"	hugepageの1枚分のページサイズ	値は"1Gi"固定
\	Hugepageの14X/Jのパークタイス	THE ACT WIFE
}		
}],		
"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
},{ "", "h"	MDCB MDC機能限を担て落停するためのですりに	上記volumeMounts."host-nvidia-mps"と同じ値
"name": "host-nvidia-mps", "host-path":	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	上高UVolumemounts. Nost-NVidia-Mps CipiU個
"hostPath": {"path": "/tmp/nvidia-mps"}		上記volumeMounts."host-nvidia-mps"の"mountPath"と同じ値
}.{		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":	400000000000000000000000000000000000000	
{"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,		ficitifue library library
"restartPolicy": "Always"		值は"Always"固定
3		
).		
{	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:1.0.0",	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	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パス	TEMPLES AND SUCH OF GROWING THE PROPERTY OF TH
"CUDA MPS LOG DIRECTORY": "/tmp/nvidia-mps",	MPSのログ出力用ディレクトリのフルパス。	
"GST PLUGIN PATH": "/opt/nvidia/deepstream/deepstream-7.0/sample-		
functions/functions/gpu infer tcp plugins/fpga depayloader",	コンテナ内でのGstreamerプラグインの格納ディレクトリ。	
"HEIGHT": "416",	入力映像のフレームサイズ(高さ)。	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)。	経験性機能なので416
Wishin, 410	ブバブラス 除い ジブバブラレー ユラース (福力)。	TAMERICAN CONTROL OF THE CONTROL OF
"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",	117,000 00000000000000000000000000000000	25) J. CERVET A DAY OFFICIAL CONTROL OF THE CONTRO
"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".		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	東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 - ev fpadepay", "%sINPUTIP%", "%sINPUTPORT%", "! video/x-raw,format=(string)BGR,%WIDTH%,%HEIGHT%", "! nvideoconvert! video/x-raw(memory:NVMM), format=(string)RGBA\", "! m.sink_0 nvstreammux name=m nvbuf-memory-type=0 batch-size=1", "%wIDTH%", "%sHEIGHT%", "! queue ! nvinfer config-file-path=./config_infer_primary_yoloV3_tiny.bt\", "batch-size=1 model-engine-file=./model_b1_gpu0_int8.engine! queue! nvvideoconvert!", "%deio/x-raw, format=(string)BGR'! videoconvert! queue! perf! rtpvrawpay! udpsink", "%GUTPUTP%", "%GUTPUTPORT%", "%OUTPUTPORT%", "securitvContext": f	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ブラグインの実行コマンドと引数
securityContext :{ "privileged": true		
privileged , dide		+
"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"	, Danielo Cializza y Greenen I y 71 y 13 y 10 at 10 at 10	環境変数"CUDA MPS PIPE DIRECTORY"と同じ値
}]		
}],		
"volumes":[{		
"name": "host-nyidia-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":		
{"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
}		1/2 1/2 1/2 1/2 1/2 1/2 1/2 1/2 1/2 1/2
}		
ที่		

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.json)と軽量推論向けのフィルタリサイズ用(fpgafunc-config-filter-resize-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た軽量推論向けに分かれている様に)入力パラメータの値ごとに別途コンフィグ情報を作成する。(同じ処理でも入力パ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)
{	Mrs-73	
"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のビットストリーム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に設定するパラメータ群
"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2140
"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"	
}	

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のビットストリーム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に設定するパラメータ群	
"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": 2140,	入力フレームの高さのサイズ	高度用でも軽量用でも2140	
"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"			

CPUFunc用コンフィグ情報

※デコード用(cpufunc-config-decode.json)と高度推論向けのフィルタリサイズ用(cpufunc-config-filter-resize-high-infer.json)と軽量推論向けのフィルタリサイズ用(cpufunc-config-filter-resize-low-infer.json)とコピー分岐用(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、入力パラメータの値ごとに別途コンフィグ情報を作成する。(入力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:1.0.0",	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",		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":{		
"privileged": true		値は"true"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	値は"hugepage-1gi"固定
"mountPath": "/dev/hugepages"		値は"/dev/hugepages"固定
},{		
"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"		値は"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"32G"固定で良い
},		
"limits":{		
	L	

III (A)II II (A)II	1 - 11-11-1 - 1 - 2011 (=F	EARL CURPS
"hugepages-1Gi": "1Gi"	hugepageの1枚分のページサイズ	值は"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
"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,		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	TTCPT固定
"imageURI": "localhost/cpu decode:1.0.0",	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"のか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",	プレベル	CPUTUTION コープジェミジュールのコンナナ夫打に必要な深見支がて、pourhナンフレート記 Pospec.containers[i].envic或定される 値は特に変更不要
"DECENV_AFFLOG_LEVEL : 0 , "DECENV FRAME WIDTH": "3840",	入力映像のフレームサイズ(幅)	聞いずいとまずで サンプルニスケースでは3840固定
"DECENV_FRAME_WIDTH : 3640 , "DECENV FRAME HEIGHT": "2160",	入力映像のフレームサイズ(高さ)	ランパエ - ハース (1630年10回定 サンブルユースケースでは2160固定
"DECENV_FRAME_HEIGHT: 2160 , "DECENV_VIDEO_CONNECT_LIMIT": "0",	・	ラファルエースクーへ Cicationing デファルト 他でいるまで良い
"DECENV_VIDEO_CONNECT_LIMIT: 0 , "DECENV_VIDEOSRC_PROTOCOL": "RTP",	映像区信元との建続技術回数。 受信するプロトコル	ファルバー しゅん CRO REPORT TO MAKE C
		KIP 向上 "TCP"固定
"DECENV_OUTDST_PROTOCOL": "TCP"	送信するプロトコル	ICP DIE
<u>},</u>	# # # # # # # # # # # # # # # # # # #	コンニュー(- 10 / 10 / 10 / 10 / 10 / 10 / 10 / 10
"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(実行ファイルのパス)
"securityContext":{		
"privileged": true		值は"true"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
}		
}		
)]		
		

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:1.0.0",	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)	サンプ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)	サンプ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)", "in_width=\$(FRENV_INPUT_WIDTH)", "in_height=\$(FRENV_INPUT_HEIGHT)", "out_height=\$(FRENV_OUTPUT_WIDTH)", "out_height=\$(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
"loglevel=\$(FRENV APPLOG LEVEL)"],		
"securityContext":{		the supplemental to the su
"privileged": true		值は"true"固定
<u>}</u>],		
"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"		値は"Always"固定
}		
}		
)]		

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:1.0.0",	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": "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
},		
"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軽量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)",	が出るのコンハン(の矢11コイン)	CPUTUILITYノリルアプリイ人文型主てノユールで、軽量推画「PJV IL 大小リック」という。これにコノアアで、大小リックにないコインド
"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
}		
}],		
"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
}		
}		
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:1.0.0",	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。タグの数字はパージョン番号(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-	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",		
"%RECEIVING%",		
"%NUM%",	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
"%FORWARDING%",		
"%MEMSIZE%"],		
"securityContext":{		
"privileged": true		値は"true"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"		値は"Always"固定
}		
}		
}]		

glue(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)
[{	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	TDMA 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:1.0.0",	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
"template":{	作成するPodのテンプレートデータ	And the state of t
"apiVersion": "v1",	11/1/2012-17/17	
"kind": "Pod",		
"spec":{		
"containers":[{	Podで起動するコンテナに関する設定情報	
"name": "cfunc-glue-fdma-to-tcp-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-	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 glue 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",	2007 9 27 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	
"%FORWARDING%",	ココンド宇仁味に渡す引数	CPUFunc用glue(DMA→TCP変換)処理モジュールの実行コマンドとその引数
"%WIDTH%",	コマンド実行時に渡す引数	CPOPulichylide(DMAマイCPを)次少主てフェールの失行コイントででいう!数
"%HEIGHT%"],		
"securityContext":{		
"privileged": true		値は"true"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	値は"hugepage-1gi"固定
"mountPath": "/dev/hugepages"		値は"/dev/hugepages"固定
}.{		
"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"		値は"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。
menory : 52G		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
}		
}		
}],		
"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
},{		
"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,		值は"true"固定
"restartPolicy": "Always"		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:1.0.0		
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_worker/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て変更するポイント)		
crc_pcieconnection_daemonset.yaml	説明	備考
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:1.0.0		
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hugepageのサイズ。基本は左記の値のままで良い。
requests:		
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		
volumes:		

4. CRCのdaemonset用YAML説明

- name: crc-pcieconnection-daemon		
hostPath:		
path: /etc/k8s_worker/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	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:1.0.0	
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_worker/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:1.0.0		
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_worker/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:1.0.0		
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_worker/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)

Function」ノトローフ用(特に環境に合わせて変更する固性は無い)		
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:1.0.0		
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_worker/config		
type: File		