



TERASOLUNA Batch Framework for Java

設定ファイル説明書

第 2.0.1.0 版

株式会社 NTT データ

本ドキュメントを使用するにあたり、以下の規約に同意していただく必要があります。同意いただけない場合は、本ドキュメント及びその複製物の全てを直ちに消去又は破棄してください。

1. 本ドキュメントの著作権及びその他一切の権利は、NTT データあるいは NTT データに権利を許諾する第三者に帰属します。
2. 本ドキュメントの一部または全部を、自らが使用する目的において、複製、翻訳、翻案することができます。ただし本ページの規約全文、および NTT データの著作権表示を削除することはできません。
3. 本ドキュメントの一部または全部を、自らが使用する目的において改変したり、本ドキュメントを用いた二次的著作物を作成することができます。ただし、「TERASOLUNA Batch Framework for Java (設定ファイル説明書)」あるいは同等の表現を、作成したドキュメント及びその複製物に記載するものとします。
4. 前2項によって作成したドキュメント及びその複製物を、無償の場合に限り、第三者へ提供することができます。
5. NTT データの書面による承諾を得ることなく、本規約に定められる条件を超えて、本ドキュメント及びその複製物を使用したり、本規約上の権利の全部又は一部を第三者に譲渡したりすることはできません。
6. NTT データは、本ドキュメントの内容の正確性、使用目的への適合性の保証、使用結果についての的確性や信頼性の保証、及び瑕疵担保義務も含め、直接、間接に被ったいかなる損害に対しても一切の責任を負いません。
7. NTT データは、本ドキュメントが第三者の著作権、その他如何なる権利も侵害しないことを保証しません。また、著作権、その他の権利侵害を直接又は間接の原因としてなされる如何なる請求(第三者との間の紛争を理由になされる請求を含む。)に関しても、NTT データは一切の責任を負いません。

本ドキュメントで使用されている各社の会社名及びサービス名、商品名に関する登録商標および商標は、以下の通りです。

Java, JDK, J2SE, J2EE は、米国 Sun Microsystems, Inc.の米国及びその他の国における登録商標または商標です。

Oracle は、米国 Oracle International Corp.の米国及びその他の国における登録商標または商標です。

TERASOLUNA は、株式会社 NTT データの登録商標です。

WebLogic は、BEA Systems Inc.の登録商標または商標です。

その他の会社名、製品名は、各社の登録商標または商標です。

本書は、TERASOLUNA Batch Framework for Java ver2.0.1.0 に対応しています。

Bean定義ファイル一覧

項番	名称	ファイル名
1	フレームワークBean定義ファイル	FrameworkBean.xml
2	ジョブBean定義ファイル	任意
3	デフォルトBean定義ファイル	DefaultValueBean.xml
4	データアクセスBean定義	dataAccessContext-batch.xml
5	非同期型ジョブ起動機能バッチデーモン	AsyncBatchDaemonBean.xml
6	デーモン終了用ジョブBean定義ファイル	StopDaemonBean.xml
7	非同期型起動用スレッドプール定義ファイル	ThreadPoolContext-AsyncBatch.xml
8	同期型起動用スレッドプール定義ファイル	ThreadPoolContext-batch.xml
9	非同期型ジョブ起動機能バッチデーモン(CommonJ用)定義ファイル	AsyncBatchDaemonForCommonjBean.xml
10	ジョブスターター用ワークマネージャ定義ファイル	WorkManagerContext.xml
11	ワーク用ワークマネージャ定義ファイル	WorkManagerTaskContext-batch.xml
12	プレースホルダ設定定義ファイル	PlaceholderConfig.xml
13	チャンク別トランザクションモデル	ChunkTransactionBean.xml
14	チャンク別トランザクションモデル(リスタート)	ChunkTransactionForRestartBean.xml
15	チャンク別トランザクションモデル(ブレイク)	ChunkTransactionForControlBreakBean.xml
16	非トランザクションモデル	NoTransactionBean.xml
17	単一トランザクションモデル	SingleTransactionBean.xml
18	ジョブ分割チャンク別トランザクションモデル	PartitionChunkTransactionBean.xml
19	ジョブ分割チャンク別トランザクションモデル(リスタート)	PartitionChunkTransactionForRestartBean.xml
20	ジョブ分割非トランザクションモデル	PartitionNoTransactionBean.xml
21	ジョブ分割単一トランザクションモデル	PartitionSingleTransactionBean.xml
22	ジョブ分割逐次単一トランザクションモデル	SequentialSingleTransactionBean.xml
23	ファイルアクセス	FileAccessBean.xml

フレームワークBean定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	Bean概要	property	Property概要
1	ブレースホルダ				PlaceholderConfig.xml	ブレースホルダを定義したBean定義ファイル
2	デフォルトBean定義ファイルのインポート				DefaultValueBean.xml	デフォルト値を定義したBean定義ファイル
3	リソース定義	messageSource	org.springframework.context.support.ResourceBundleMessageSource	リソースバンドルのBean定義。	basenames	リソースの設定。 初期設定値は『ApplicationResources.system-messages.application-messages』である。
4		messageAccessor	jp.terasoluna.fw.batch.springsupport.messages.MessageAccessorImpl	メッセージ取得用クラスの設定	-	-
5	分割キー取得用コレクタ定義	IBatisDbPartitionKeyCollector	jp.terasoluna.fw.batch.ibatissupport.IBatisDbCollectorImpl	データベースCollectorのPartitionKey取得用Bean定義。 ジョブBean定義ファイルでCollectorに設定されたDAOを使って、データベースからPartitionKeyデータを取得する。	queryDAO	使用するQueryDao。 初期設定値は『queryDAO』である。
6		listPropertyPartitionKeyCollector	jp.terasoluna.fw.batch.standard.ListPropertyCollector	Bean定義リストCollectorのPartitionKey取得用Bean定義。 ジョブBean定義ファイルに設定されたリストを使ってPartitionKeyデータを取得する。	collectedDataHandlerFactory	分割キー処理用ハンドラのファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.partition.PartitionKeyHandlerFactory』である。
7		stringArrayPropertyPartitionKeyCollector	stringArrayPropertyCollector	文字列配列プロパティCollectorのPartitionKey取得用Bean定義。 ジョブBean定義ファイルに設定された文字列配列を使って、PartitionKeyデータを取得する。	collectedDataHandlerFactory	分割キー処理用ハンドラのファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.partition.PartitionKeyHandlerFactory』である。
8	対象データ取得用コレクタ定義	listPropertyCollector	jp.terasoluna.fw.batch.standard.ListPropertyCollector	Bean定義リストCollector用Bean定義。 ジョブBean定義ファイルに設定されたリストを使って対象データを取得する。	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.standard.ChunkerFactory』である。
9		stringArrayPropertyCollector	stringArrayPropertyCollector	文字列配列プロパティCollector用Bean定義。 ジョブBean定義ファイルに設定された文字列配列を使って対象データを取得する。	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.standard.ChunkerFactory』である。
10		IBatisDbChunkCollector	jp.terasoluna.fw.batch.ibatissupport.IBatisDbCollectorImpl	データベースCollector用Bean定義。 ジョブBean定義ファイルでCollectorに設定されたDAOを使って、データベースから対象データを取得する。	queryDAO	使用するQueryDao。 初期設定値は『queryDAO』である。
			jp.terasoluna.fw.batch.standard.ChunkerFactory	チャンカー作成クラスの定義。	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.standard.ChunkerFactory』である。
					chunkSize	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
11	入力チェック対象データ取得用コレクタ定義	validationListPropertyCollector	jp.terasoluna.fw.batch.standard.ListPropertyCollector	Bean定義リストCollector用Bean定義。 ジョブBean定義ファイルに設定されたリストを使って対象データを取得する。	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.validation.ValidationExecutorFactory』である。
12		validationStringArrayPropertyCollector	stringArrayPropertyCollector	文字列配列プロパティCollector用Bean定義。 ジョブBean定義ファイルに設定された文字列配列を使って対象データを取得する。	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.validation.ValidationExecutorFactory』である。
13		validationIBatisDbChunkCollector	jp.terasoluna.fw.batch.ibatissupport.IBatisDbCollectorImpl	データベースCollector用Bean定義。 ジョブBean定義ファイルでCollectorに設定されたDAOを使って、データベースから対象データを取得する。	queryDAO	使用するQueryDao。 初期設定値は『queryDAO』である。
			jp.terasoluna.fw.batch.validation.ValidationExecutorFactory	入力チェック用の対象データ取得ハンドラ (CollectedDataHandler) クラスの定義。 チャンク作成処理は、collectedDataHandlerFactoryに設定されたチャンカー生成	collectedDataHandlerFactory	チャンカーを生成するファクトリクラス。 初期設定値は『jp.terasoluna.fw.batch.validation.ValidationExecutorFactory』である。
14					validator	バリデータの定義。
15					validationResultHandler	入力チェック用XML設定ファイルのパスの定義。
16	マネージャ定義	baseManager	jp.terasoluna.fw.batch.core.JobManager	ジョブ等のフレームワークで規定する作業単位を管理、実行するマネージャ用Bean定義。	collectorResultHandler	対象データ取得結果のハンドラBeanの定義。 初期設定値は『CollectorResultHandler』である。
					exceptionHandlerMap	例外ハンドラのBean定義。 初期設定値は『exceptionHandlerMap』である。
					defaultJobExceptionHandler	デフォルト例外ハンドラ用Bean定義。 初期設定値は『defaultJobExceptionHandler』である。
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
17		baseWorker	jp.terasoluna.fw.batch.core.JobWorker	Chunkを処理単位として、Chunkが持つデータを入力とするビジネスロジックを実行するマネージャ用Bean定義。	batchUpdateResultHandler	バッチ更新の処理結果ハンドラ用Bean定義。 初期設定値は『batchUpdateResultHandler』である。
					defaultJobExceptionHandler	デフォルト例外ハンドラ用Bean定義。 初期設定値は『defaultJobExceptionHandler』である。
					exceptionHandlerMap	例外ハンドラを格納したMapのBean定義。 初期設定値は『exceptionHandlerMap』である。
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
18		baseQueueProcessor	jp.terasoluna.fw.batch.standard.QueueProcessor	キューから要素を取り出し、取り出したそれぞれを要素をパラメータとしてワーカを起動するマネージャ用Bean定義。	なし	
19	サポートプロセッサ	transactionalSupportProcessor	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalSupportProcessor	トランザクション処理用Bean定義。 SupportProcessorインタフェースの実装クラス。	supportLogicResultHandler	サポートロジックの処理結果ハンドラ用設定。 初期設定値は『transactionalSupportLogicResultHandler』である。
					defaultJobExceptionHandler	デフォルト例外ハンドラの設定。 初期設定値は『defaultJobExceptionHandler』である。
					exceptionHandlerMap	例外ハンドラマップの設定。 初期設定値は『exceptionHandlerMap』である。
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
20		standardSupportProcessor	jp.terasoluna.fw.batch.standard.StandardSupportProcessor	サポートロジックの起動用クラスのBean定義。	supportLogicResultHandler	サポートロジックの処理結果ハンドラ用設定。 初期設定値は『supportLogicResultHandler』である。
					defaultJobExceptionHandler	デフォルト例外ハンドラの設定。 初期設定値は『defaultJobExceptionHandler』である。
					exceptionHandlerMap	例外ハンドラマップの設定。 初期設定値は『exceptionHandlerMap』である。
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。

ジョブBean定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要	必須
1	処理モデルインポート				ChunkTransactionBean.xml	チャンク別トランザクションモデルBean定義ファイル	○ 処理モデルを一つ選び、インポートする必要がある
2					ChunkTransactionForRestartBean.xml	リスタート機能を使用するチャンク別トランザクションモデルBean定義ファイル	
3					ChunkTransactionForControlBreakBean.xml	コントロールブレイク機能を使用するチャンク別トランザクションモデルBean定義ファイル	
4					NoTransactionBean.xml	トランザクション制御なしモデルのBean定義ファイル	
5					PartitionChunkTransactionBean.xml	分割ジョブのチャンク別トランザクションモデルBean定義ファイル	
6					PartitionChunkTransactionForRestartBean.xml	リスタート機能を使用する分割ジョブのチャンク別トランザクションモデルBean定義ファイル	
7					PartitionNoTransactionBean.xml	分割ジョブのトランザクション制御なしモデルのBean定義ファイル	
8					PartitionSingleTransactionBean.xml	分割ジョブの単一トランザクションモデルのBean定義ファイル	
9					SingleTransactionBean.xml	単一トランザクションモデルのBean定義ファイル	
10					SequentialSingleTransactionBean.xml	ジョブ分割逐次単一トランザクションモデルのBean定義ファイル	
11					FileAccessBean.xml	ファイルアクセス用Bean定義ファイル	
12	親ジョブ前処理	parentJobPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	分割ジョブ実行時の親ジョブの前処理設定	sourceList	前処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した前処理から実行される。	任意
13	親ジョブ後処理	parentJobPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	分割ジョブ実行時の親ジョブの後処理の設定	sourceList	後処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した後処理から実行される。	任意
14	分割キー取得	partitionKeyCollector	フレームワークBean定義ファイル説明書の分割キー処理用コレクタ定義から選択。	分割ジョブ実行時の分割キー取得処理定義	任意のプロパティ	『sql』の場合は、IBATISのSQL定義ファイルで定義されたSQL IDを設定する。	分割ジョブの場合は必須
15	対象データ取得	collector	任意のクラス名	対象データ取得用Beanの設定	任意のプロパティ		○
16	ビジネスロジック	blagic	ビジネスロジックの実装クラス	ビジネスロジックのBeanの設定	任意のプロパティ		○
17	ジョブコンテキスト	jobContext	パラメータ値等を格納するクラス	パラメータ値等を格納するクラスの設定	任意のプロパティ		○
18	ジョブ前処理	jobPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ実行時のジョブの前処理設定	sourceList	前処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した前処理から実行される。	任意
19	ジョブ後処理	jobPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ実行時のジョブの後処理設定	sourceList	後処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した後処理から実行される。	任意
20	先頭チャンク前処理	firstChunkPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ実行時の先頭チャンク前処理設定	sourceList	前処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した前処理から実行される。	任意
21	最終チャンク後処理	lastChunkPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ実行時の最終チャンク後処理設定	sourceList	後処理として実行するクラスをリストに定義する。複数定義した場合は、上位に定義した後処理から実行される。	任意
22	セーブポイント	useSavepoint	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	セーブポイント使用有無設定	staticField	使用する場合は『java.lang.Boolean.TRUE』を設定	任意
23	チャンクサイズ	chunkSize	java.lang.Integer	チャンクサイズ設定	-		任意
24	分割ジョブの多重起動数	multiplicity	java.lang.Integer	分割ジョブの多重起動数設定	-		分割ジョブの場合は必須
25	SqLMapConfig	sqLMapConfigFileMap	java.lang.String	SqLMapConfigの設定	任意のファイル	ジョブ毎にSqLMapを用意する場合に設定する。	任意
26	コントロールブレイク定義情報リスト	controlBreakDefinitionList	org.springframework.beans.factory.config.ListFactoryBean	コントロールブレイクのブレイク定義情報の設定	sourceList	コントロールブレイクの定義情報をリストに定義する。複数定義した場合は、上位から処理される。リストに定義する内容については項番27, 28を参照。	任意 (コントロールブレイクに対応した雛形をインポートする)
27				コントロールブレイクのブレイクキーおよびブレイクキーに対応するハンドラの設定	breakKey	コントロールブレイクのブレイクキーをリストに定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)
28			jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem		controlBreakHandler	ブレイクキーに対応したコントロールブレイクハンドラを定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)
29	チャンクコントロールブレイク定義情報	chunkControlBreakDefinitionItem		チャンクコントロールブレイクのブレイクキーおよびブレイクキーに対応するハンドラの設定	breakKey	チャンクコントロールブレイクのブレイクキーをリストに定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)
30			jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem		controlBreakHandler	ブレイクキーに対応したコントロールブレイクハンドラを定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)
31	トランスチャンクコントロールブレイク定義情報リスト	transControlBreakDefinitionList	org.springframework.beans.factory.config.ListFactoryBean	トランスチャンクコントロールブレイクのブレイク定義情報の設定	sourceList	トランスチャンクコントロールブレイクの定義情報をリストに定義する。複数定義した場合は、上位から処理される。リストに定義する内容については項番32, 33を参照。	任意 (コントロールブレイクに対応した雛形をインポートする)
32	トランスチャンクコントロールブレイク定義情報リスト			トランスチャンクコントロールブレイクのブレイクキーおよびブレイクキーに対応するハンドラの設定	breakKey	トランスチャンクコントロールブレイクのブレイクキーをリストに定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)
33			jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem		controlBreakHandler	ブレイクキーに対応したコントロールブレイクハンドラを定義する。	任意 (コントロールブレイクに対応した雛形をインポートする)

デフォルトBean定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	Bean概要	property	Property概要
1	ジョブ終了監視用クラス定義	endFileChecker	jp.terasoluna.fw.batch.init.EndFileChecker	ジョブ終了制御用Bean定義。	endFileDir	終了ファイル検出ディレクトリパス。初期設定値は『batchapps/BE-03/EndFile』である。
2		scheduledTask	org.springframework.scheduling.timer.ScheduledTimerTask	スケジューラタスク用Bean定義。Common対応機能を使用する際には『WorkManagerTaskContext-batch.xml』の設定が有効になるため『DefaultValueBean.xml』の設定は無効になる。	period timerTask	起動周期。初期設定値は『8000』である。スケジューラ起動対象。初期設定値は『endFileCheck』である。
3	タイマ	timerFactory	org.springframework.scheduling.timer.TimerFactoryBean	監視用タイマファクトリ用Bean定義。Common対応機能を使用する際には『WorkManagerTaskContext-batch.xml』の設定が有効になるため『DefaultValueBean.xml』の設定は無効になる。	daemon scheduledTimerTasks	初期設定値は『false』である。スケジューラタスク用設定。初期設定値は『scheduledTask』である。
4	デフォルト定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	useCache	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	BeanFactoryのキャッシュ有無のデフォルト設定。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	staticField	初期設定値は『java.lang.Boolean.FALSE』である。
5		parentJobPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ分割前処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
6		parentJobPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ分割後処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
7		jobPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ前処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
8		jobPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	ジョブ後処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
9		firstchunkPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	先頭チャンク前処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
10		lastchunkPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	最終チャンク後処理のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceList	初期設定値は『空白リスト』である。
11		chunkSize	java.lang.Integer	チャンクサイズ用Bean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	-	初期設定値は『20』である。
12		useSavepoint	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	セーブポイント有無のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	staticField	初期設定値は『java.lang.Boolean.FALSE』である。
13		jobStatusSummarizer	jp.terasoluna.fw.batch.standard.StandardJobStatusSummarizer	処理状態の結果ハンドラのデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	-	-
14		exitCodeMap	org.springframework.beans.factory.config.MapFactoryBean	終了コードのデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sourceMap	初期設定は以下の通り。 Key:ENDING_NORMALLY value:0 Key:ENDING_ABNORMALLY value:100 Key:SUSPENDING value:200
15		useMonitorable	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	ジョブ監視有無のデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	staticField	初期設定値は『java.lang.Boolean.FALSE』である。
16		manageableJobSize	java.lang.Integer	ジョブ監視の上限値用Bean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	-	初期設定値は『20』である。
17		sqlMapConfigFileName	java.lang.String	SqlMapConfigのデフォルトBean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	sqlMapConfigFileName	初期設定値は『common/sql-map-config.xml』である。
18		chunkQueueLength	java.lang.Integer	チャンクキューのサイズ用Bean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	-	初期設定値は『10』である。
19		partitionQueueLength	java.lang.Integer	分割キーキューのサイズ用Bean定義。ジョブBean定義に設定がある場合、ジョブBean定義が優先される。	-	初期設定値は『10』である。
20	例外ハンドラ	exceptionHandlerMap	org.springframework.beans.factory.config.MapFactoryBean	例外ハンドラマップのBean定義。	sourceMap	ハンドラマップ値の設定。初期設定値は『CollectorException=jp.terasoluna.batch.sample.bc01.CollectorExceptionHandlerImpl JobException=jp.terasoluna.fw.batch.standard.StandardJobExceptionHandler』である。
21	デフォルト例外ハンドラ定義	defaultJobExceptionHandler	jp.terasoluna.fw.batch.standard.StandardJobExceptionHandler	JobException用デフォルト例外ハンドラのBean定義。	なし	
22	Throwableハンドラ定義	throwableHandler	jp.terasoluna.fw.batch.standard.StandardThrowableHandler	ThrowableハンドラのBean定義。	なし	
23	処理結果ハンドラ	BLogicResultHandler	jp.terasoluna.fw.batch.standard.StandardBLogicResultHandler	ビジネスロジック結果ハンドラBean定義。BLogicResultHandlerの実装クラス。	なし	
24		TransactionalBLogicResultHandler	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalBLogicResultHandler	チャンク別トランザクションモデル用ビジネスロジック結果ハンドラBean定義。BLogicResultHandlerの拡張クラス。	なし	
25		CollectorResultHandler	jp.terasoluna.fw.batch.standard.StandardCollectorResultHandler	Collector結果ハンドラBean定義。StandardBLogicResultHandlerの実装クラス。	なし	
26		batchUpdateResultHandler	jp.terasoluna.fw.batch.standard.StandardBatchUpdateResultHandler	バッチ更新処理結果ハンドラBean定義。BatchUpdateResultHandlerの実装クラス。	なし	
27		supportLogicResultHandler	jp.terasoluna.fw.batch.standard.StandardSupportLogicResultHandler	サポート処理実行結果ハンドラBean定義。SupportLogicResultHandlerの実装クラス。	なし	
28	入力チェック	validationConfigLocations	org.springframework.beans.factory.config.ListFactoryBean	入力チェック定義ファイルとルールファイルのBean定義。	sourceList	初期値は『/common/validator-rules.xml /common/validator-rules-ex.xml /common/validationCommon.xml』である。
29		validationResultHandler	jp.terasoluna.fw.batch.validation.StandardValidationResultHandler	入力チェック処理結果ハンドラBean定義。ValidationResultHandlerの実装クラス。	なし	

データアクセスBean定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	Bean概要	property	Property概要
1	データソース	dataSource	org.apache.commons.dbcp.BasicDataSource等	DataSourceのBean定義。 DBCPによるコネクションプーリングを行う。	driverClassName	JDBCのドライバの設定。
					url	接続先のURLの設定。
					username	ユーザアカウントの設定。
					password	パスワードの設定。
					poolPreparedStatements	PreparedStatementsのプーリング有無（デフォルトはtrue）
					defaultAutoCommit	オートコミットの有無（デフォルトはfalse）
2	トランザクションマネージャ	transactionManager	org.springframework.jdbc.datasource.DataSourceTransactionManager	単一のJDBCデータソース向けのトランザクションマネージャBean定義。	dataSource	使用するデータソースBeanの設定。 初期設定値は『dataSource』である。
					transactionManager	トランザクションマネージャの設定。 プロキシ対象のBean、およびトランザクション定義情報の設定は、このbaseTransactionProxyを親定義とするフレームワークのBean定義で行われている。 初期設定値は『transactionManager』である。
3	トランザクションプロキシ	baseTransactionProxy	org.springframework.transaction.interceptor.TransactionProxyFactoryBean	トランザクションプロキシのベース定義となる親Bean定義。 Worker定義のtarget属性に指定するBean。子のBean定義で属性をオーバーライドすることができる。 Springで提供されているファクトリクラスのBean定義。 任意のクラスにトランザクション制御コードを付加したプロキシクラスを生成する。	transactionManager	トランザクションマネージャの設定。 プロキシ対象のBean、およびトランザクション定義情報の設定は、このbaseTransactionProxyを親定義とするフレームワークのBean定義で行われている。 初期設定値は『transactionManager』である。
4	SQLMapClient	sqlMapClient	org.springframework.orm.ibatis.SqlMapClientFactoryBean	iBATIS データベース層のためのSQLMapのBean定義。	configLocation	使用するSQLMapの定義ファイルの設定。 初期設定値は『sqlMapConfigFileName』である。（デフォルトBean定義ファイル参照）
					dataSource	使用するデータソースBeanの設定。 初期設定値は『dataSource』である。
5	DAO	queryDAO	jp.terasoluna.fw.dao.ibatis.QueryDAOiBatisImpl	参照系のDAOのBean定義。	sqlMapClient	使用するSQLMapClientの設定。 初期設定値は『sqlMapClient』である。
6		updateDAO	jp.terasoluna.fw.dao.ibatis.UpdateDAOiBatisImpl	更新系のDAOのBean定義。	sqlMapClient	使用するSQLMapClientの設定。 初期設定値は『sqlMapClient』である。
7		spDAO	jp.terasoluna.fw.dao.ibatis.StoredProcedureDAOiBatisImpl	ストアドプロシジャ用のDAOのBean定義。	sqlMapClient	使用するSQLMapClientの設定。 初期設定値は『sqlMapClient』である。

非同期型ジョブ起動機能パッチデーモン

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	非同期型ジョブ起動機能パッチデーモン	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	非同期型ジョブ起動機能パッチデーモンの定義。	name collector workQueueFactory preProcessor postProcessor	初期値 [AsyncBatchDaemonManager] 項目16を参照 項目2を参照 項目3を参照 項目5を参照
2		-	jp.terasoluna.fw.batch.standard.StandardDilloQueueFactory	非同期型ジョブ起動機能パッチデーモンの定義。	workerCalculatorService queueProcessor queueLength multiplicity	フレームワークBean定義ファイルを参照 項目5を参照 初期値 [10] 項目29を参照
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	supportProcessor	項目4を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	-	-
5		parentJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	supportProcessor	項目6を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	-	-
7		jobContext	jp.terasoluna.fw.batch.init.RootJobContext	親ジョブ用ジョブコンテキストの定義。	-	-
8	キュープロセッサ	asyncBatchDaemonQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	name worker	初期値 [AsyncBatchDaemonQueueProcessor] 項目9を参照
9	パッチ起動用ワーク	childJobManager	jp.terasoluna.fw.batch.spring.support.init.JobExecutor	ジョブ起動用クラスの定義。	constructor-arg async preJobProcessor postJobProcessor throwableHandler	Bean定義ファイル 初期値 [/common/FrameworkBean.xml, /common/dataAccessContext-batch.xml, /common/ThreadPoolContext-AsyncBatch.xml] 非同期型起動有無フラグ 初期値 [true] 項目10を参照 項目13を参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照。
10		preJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	supportProcessor	項目11を参照
11		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	supportLogicList	項目12を参照
12		-	jp.terasoluna.fw.batch.spring.support.init.AsyncJobPreLogic	非同期型ジョブ前処理の定義。	jobControlTableHandler	フレームワークBean定義ファイルを参照
13		postJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	supportProcessor	項目14を参照
14		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	項目15を参照
15		-	jp.terasoluna.fw.batch.spring.support.init.AsyncJobPostLogic	非同期型ジョブ後処理の定義。	jobControlTableHandler	フレームワークBean定義ファイルを参照
16	コレクター	batchDaemonJobCollector	jp.terasoluna.fw.batch.init.JobRequestInfoCollector	非同期型ジョブ起動時のジョブ依頼情報取得用クラスの定義。	collectedDataHandlerFactory jobControlTableHandler	項目17を参照 フレームワークBean定義ファイルを参照
17		-	jp.terasoluna.fw.batch.init.JobRequestInfoHandlerFactory	ジョブ依頼情報格納用ワークキュー生成用クラスの定義。	intervalSeconds	ジョブ管理テーブルの監視周期 (秒) 初期値 [10]
18	ジョブの処理状況クラス定義	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister	ジョブ監視用Bean定義 初期設定値は [MBeanRegister] であり、監視を行わない場合は設定する必要はない。
19		-	-	transactionManager	トランザクションマネージャのBean定義。 データアクセスBean定義ファイルを参照。	
20		-	-	useSavepoint	セーブポイント有無のBean定義。 デフォルトBean定義ファイルを参照。	
21		-	-	partitionNo	ジョブパーティション番号のBean定義。 初期設定値は [-1] である。	
22	ジョブの処理状況クラス	JobStatus	jp.terasoluna.fw.batch.spring.support.transaction.TransactionalJobStatus	ジョブの処理状況用Bean定義。	transactionManager	トランザクションマネージャのBean定義。 データアクセスBean定義ファイルを参照。
23		-	-	useSavepoint	セーブポイント有無のBean定義。 デフォルトBean定義ファイルを参照。	
24		-	-	partitionNo	ジョブパーティション番号のBean定義。 初期設定値は [-1] である。	
25	ジョブ監視	useMonitorable	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	ジョブ監視有無のBean定義。	staticField	初期値 [java.lang.Boolean.FALSE]
26	ジョブ管理テーブル	jobControlTableHandler	jp.terasoluna.fw.batch.init.JobControlTableHandlerImpl	ジョブ管理テーブル用ハンドラのBean定義。	queryDAO updateDAO	参照系のDAOのBean定義。 データアクセスBean定義ファイルを参照。 更新系のDAOのBean定義。 データアクセスBean定義ファイルを参照。
27		-	-	-	-	-
28	スレッド管理	threadSize	jp.terasoluna.fw.batch.spring.support.standard.ThreadSizeFactoryBean	スレッドプール作成時のサイズ	multiplicity	項目29を参照。
29		multiplicity	java.lang.Integer	多重度	constructor-arg	初期値 [1]

デーモン終了用ジョブBean定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	非同期バッチ起動用のマネージャ	jobManager	jp.terasoluna.fw.batch.init.NullWorker	非同期バッチ起動用のマネージャ	-	-
2		jobContext	jp.terasoluna.fw.batch.springsupport.init.AsyncJobContext	ジョブコンテキスト	-	-
3	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister	ジョブ監視用Bean定義 初期設定値は『MBeanRegister』であり、監視を行わない場合は設定する必要はない。
4					partitionNo	ジョブパーティション番号のBean定義。 初期設定値は『-1』である。
5		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo	ジョブパーティション番号のBean定義。 初期設定値は『-1』である。

非同期型起動用スレッドプール定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	Bean概要	property	Property概要
1	スレッドプール管理	workerExecutorService	jp.terasoluna.fw.batch.springsupport.standard.FixedThreadPoolWorkerExecutorService	フレームワークが利用するスレッドプールのBean定義。	constructor-arg defaultJobExceptionHandler exceptionHandlerMap throwableHandler	スレッド数の設定。 初期設定値は『20』である。 デフォルト例外ハンドラの設定。 初期設定値は『defaultJobExceptionHandler』である。 例外ハンドラマップの設定。 初期設定値は『exceptionHandlerMap』である。 ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。

同期型起動用スレッドプール定義ファイル

順番	設定項目名称	Bean ID	Classes/parent名	Bean概要	property	Property概要
1	スレッドプール管理	workerExecutorService	jp.terasoluna.fw.batch.springsupport.standard.FixedThreadPoolWorkerExecutorService	フレームワークが利用するスレッドプールのBean定義。	constructor-arg defaultJobExceptionHandler exceptionHandlerMap throwableHandler	スレッド数の設定。 初期設定値は『threadSize』である。 デフォルト例外ハンドラの設定。 初期設定値は『defaultJobExceptionHandler』である。 例外ハンドラマップの設定。 初期設定値は『exceptionHandlerMap』である。 ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。

非同期型ジョブ起動機能バッチデーモン(CommonJ用)

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	非同期バッチ起動用マネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	非同期バッチ起動用マネージャの定義。	name collector workQueueFactory preProcessor postProcessor	初期値【AsyncBatchDaemonManager】 項番16を参照 項番2を参照 項番3を参照 項番5を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	非同期バッチ起動用作業キュー作成クラスの定義 (CommonJ用)。	workerCalculatorService QueueProcessor	フレームワークBean定義ファイルを参照 項番5を参照 初期値【10】 multiplicity 項番29を参照
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transactionalSupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	supportProcessor	項番4を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	-	-
5		parentJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transactionalSupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	supportProcessor	項番6を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	-	-
7		jobContext	jp.terasoluna.fw.batch.init.RootJobContext	親ジョブ用ジョブコンテキストの定義。	-	-
8	キュープロセッサ	asyncBatchDaemonQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	name worker	初期値【AsyncBatchDaemonQueueProcessor】 項番9を参照
9	バッチ起動用ワーク	childJobManager	jp.terasoluna.fw.batch.common.init.JobExecutor	ジョブ起動用クラスの定義。	constructor-arg async preJobProcessor postJobProcessor jobParametersSplitStr throwableHandler	Bean定義ファイル 初期値 【 /common/FrameworkBean.xml./common/dataAccessContext-batch.xml./common/WorkManagerTaskContext-batch.xml./common/ValidationContext-AsyncBatch.xml】 非同期起動有無フラグ 初期値【true】 項番10を参照 項番13を参照 ジョブパラメータの区切り文字 初期設定値は【,】である。 ThrowableハンドラのBean定義。 初期設定値は【throwableHandler】である。 デフォルトBean定義ファイルを参照。
10		preJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transactionalSupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	supportProcessor	項番11を参照
11		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	supportLogicList	項番12を参照
12		-	jp.terasoluna.fw.batch.springsupport.init.AsyncJobPreLogic	非同期用ジョブ前処理の定義。	JobControlTableHandler	フレームワークBean定義ファイルを参照
13		postJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transactionalSupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	supportProcessor	項番14を参照
14		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	項番15を参照
15		-	jp.terasoluna.fw.batch.springsupport.init.AsyncJobPostLogic	非同期用ジョブ後処理の定義。	JobControlTableHandler	フレームワークBean定義ファイルを参照
16	コレクター	batchDaemonJobCollector	jp.terasoluna.fw.batch.init.JobRequestInfoCollector	非同期ジョブ起動時のジョブ依頼情報取得用クラスの定義。	collectedDataHandlerFactory jobControlTableHandler	項番17を参照 フレームワークBean定義ファイルを参照
17		-	jp.terasoluna.fw.batch.init.JobRequestInfoHandlerFactory	ジョブ依頼情報格納用ワークキュー生成用クラスの定義。	intervalSeconds refreshCount	ジョブ管理テーブルの監視周期(秒) 初期値【10】 初期値【20】
18	ジョブの処理状況クラス定義	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableTransactionalJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister	ジョブ監視用Bean定義 初期設定値は【MBeanRegister】であり、監視を行わない場合は設定する必要はない。
19		-	-	transactionManager	トランザクションマネージャのBean定義。 データアクセスBean定義ファイルを参照。	
20		-	-	useSavepoint	セーブポイント有無のBean定義。 デフォルトBean定義ファイルを参照。	
21	ジョブの処理状況クラス	JobStatus	jp.terasoluna.fw.batch.springsupport.transactionalJobStatus	ジョブの処理状況用Bean定義。	partitionNo	ジョブパーティション番号のBean定義。 初期設定値は【-1】である。
22		-	-	transactionManager	トランザクションマネージャのBean定義。 データアクセスBean定義ファイルを参照。	
23		-	-	useSavepoint	セーブポイント有無のBean定義。 デフォルトBean定義ファイルを参照。	
24	ジョブ監視	useMonitorable	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	ジョブ監視有無のBean定義。	partitionNo	ジョブパーティション番号のBean定義。 初期設定値は【-1】である。
25		-	-	staticField	初期値【java.lang.Boolean.FALSE】	
26	ジョブ管理テーブル	jobControlTableHandler	jp.terasoluna.fw.batch.init.JobControlTableHandlerImpl	ジョブ管理テーブル用ハンドラのBean定義。	queryDAO	参照系のDAOのBean定義。 データアクセスBean定義ファイルを参照。
27		-	-	updateDAO	更新系のDAOのBean定義。 データアクセスBean定義ファイルを参照。	
28	スレッド管理	threadSize	jp.terasoluna.fw.batch.springsupport.standard.ThreadSizeFactoryBean	スレッドプール作成時のサイズ	multiplicity	項番29を参照。
29		-	-	-	constructor-arg	初期値【3】

ジョブスターター用ワークマネージャ定義ファイル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	ジョブスターター	jobStarter	jp.terasoluna.fw.batch.commonj.init.JobStarterImpl	ジョブ起動の定義	workManager workListener	項番2を参照。 初期値は「manager」である。 項番3を参照。 初期値は「listener」である。
2	ワークマネージャ	manager	org.springframework.scheduling.commonj.WorkManagerTaskExecutor	ワークマネージャの定義	workManagerName resourceRef	weblogic-application.xml に記述される、WorkManager定義のname属性を参照。 初期値は「wm/BatchWorkManager」である。 初期値は「true」である。
3	ワークリスナー	listener	jp.terasoluna.fw.batch.commonj.listener.JobExecutorListener	ワークリスナーの定義	jobResultInfoHandler	項番6を参照。
4	SqlMapConfigファイルPath	sqlMapConfigFileName	java.lang.String	SqlMapConfigファイルのパス定義	constructor-arg	使用するSqlMapConfigの定義ファイルの設定。 初期値は「common/sql-map-config.xml」である。
5	ジョブ管理テーブル用ハンドラ	jobMessageTableHandler	jp.terasoluna.fw.batch.jobmessage.JobMessageTableHandlerImpl	ジョブ管理テーブル用ハンドラの定義	queryDAO updateDAO	参照系のDAOのBean定義。 初期値は「queryDAO」である。 更新系のDAOのBean定義。 初期値は「updateDAO」である。
6	ジョブ結果テーブル用ハンドラ	jobResultInfoHandler	jp.terasoluna.fw.batch.commonj.transaction.JobResultInfoHandlerImpl	ジョブ結果テーブル用ハンドラの定義	updateDAO transactionManager	参照系のDAOのBean定義。 初期値は「updateDAO」である。 データアクセスBean定義を参照。 初期値は「transactionManager」である。

ワーク用ワークマネージャ定義ファイル

順番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	ワーク管理	workExecutorService	jp.terasoluna.fw.batch.commonj.WorkManagerTaskExecutorService	ワーク管理の定義	defaultJobExceptionHandler	デフォルトBean定義ファイルを参照。 初期値は『defaultJobExceptionHandler』である。
					exceptionHandlerMap	デフォルトBean定義ファイルを参照。 初期値は『exceptionHandlerMap』である。
					workManager	項番2を参照。 初期値は『TaskExecutor』である。
2	taskExecutor	org.springframework.scheduling.commonj.WorkManagerTaskExecutor	タスク実行の定義	workManagerName	weblogic-application.xmlに記述される、WorkManager定義のname属性を参照。 初期値は『wm.BatchWorkManager』である。	
				resourceRef	初期値は『true』である。	
3	workListener	jp.terasoluna.fw.batch.commonj.listener.WorkQueueListener	ワークリスナーの定義	-	-	
4	scheduledTask	org.springframework.scheduling.commonj.ScheduledTaskListener	スケジューラの定義 Commonj対応機能を使用するには 『WorkManagerTaskContext-batch.xml』の設定が有効になるため『DefaultValueBean.xml』の設定は無効になる。	delay	遅延時間の設定。 初期値は『0』である。	
				period	繰り返す周期の設定。 初期値は『8000』である。(単位は『ms』)	
				fixedRate	初期値は『false』である。	
				runnable	デフォルトBean定義ファイルを参照。 初期値は『endFileChecker』である。	
5	監視	timerFactory	org.springframework.scheduling.commonj.TimerManagerFactoryBean	タイマーの定義 Commonj対応機能を使用するには 『WorkManagerTaskContext-batch.xml』の設定が有効になるため『DefaultValueBean.xml』の設定は無効になる。	timerManagerName	weblogic-application.xmlに記述される、WorkManager定義のname属性を参照。 初期値は『timer/FileCheckTimer』である。
					resourceRef	初期値は『true』である。
					shared	初期値は『true』である。 タイマーはシングルトンである必要があるため、必ずtrueである必要がある。
					scheduledTimerListeners	タイマーリスナーのBean定義。 listタグにより複数のリスナーを設定できる。 初期値は『scheduledTask』である。
					jndiTemplate	JNDI環境設定。 初期値は『jndiTemplateForTimerManager』である。

プレースホルダ設定定義ファイル

順番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	プレースホルダ	-	org.springframework.beans.factory.config.PropertyPlaceholderConfigurer	ジョブが使用するプレースホルダの設定定義。	locations	使用するプレースホルダのパスを定義する。 初期値は 『 jdbc.properties 』 『 template/workQueueFactory.properties 』

チャンク別トランザクションモデル

順番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ					
1		jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	ブレースホルダの定義 ジョブ実行マネージャの定義。	import resource name collector workQueueFactory preProcessor postProcessor	初期値 [classpath:common/PlaceholderConfig.xml] 初期値 [chunkTransactionJobManager] ジョブBean定義ファイルを参照 項番2を参照 項番3を参照 項番3を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応作業キュー生成クラスの定義。	workerExecutorsService queueLength queueProcessor	同期型起動用スレッドプール定義ファイルを参照 非同同期型起動用スレッドプール定義ファイルを参照 デフォルトBean定義ファイルを参照 項番7を参照
3	マネージャ	jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	supportProcessor	項番4を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
5		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ後処理の定義。	supportProcessor	項番6を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
7	キュープロセッサ	chunkTransactionQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	キュー処理クラスの定義。	name worker	初期値 [chunkTransactionQueueProcessor] 項番8を参照
8		chunkTransactionWorker	jp.terasoluna.fw.batch.spring.support.transactionalTransactionWorker jp.terasoluna.fw.batch.core.JobWorker	トランザクション制御有りのワーカー定義。 チャンク処理クラスの定義。	jobWorker	項番9を参照 項番10を参照 項番11を参照
9	ワーカー	jobWorker	jp.terasoluna.fw.batch.core.StandardLogicExecutor	ビジネスロジック実行クラスの定義。	bloticExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler throwableHandler	デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowbleハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
10		bloticExecutor	jp.terasoluna.fw.batch.core.StandardLogicExecutor	ビジネスロジック実行クラスの定義。	blotic bloticResultHandler exceptionHandlerMap defaultJobExceptionHandler throwableHandler	ジョブBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowbleハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
11		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	updateDAO	デフォルトBean定義ファイルを参照 データアクセスBean定義ファイルを参照
12	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister transactionManager useSavepoint partitionNo	フレームワークBean定義ファイルを参照 データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [1]
13		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo transactionManager useSavepoint	初期値 [1] データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照
14	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	constructor-arg	初期値 [2]

チャック別トランザクションモデル (リスタート)

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
マネージャ	1	jobManager	jp.terasoluna.fw.batch.restart.JobContextRestorer	リスタート実行用マネージャ定義。	jobRestartTableHandler	項番21を参照
					jobManager	項番2を参照
	2	delegatedJobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	ジョブ実行マネージャの定義。	name	初期値 [chunkTransactionJobManager]
					collector	ジョブBean定義ファイルを参照
	3	-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common)対応作業キュー生成クラスの定義。	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 後期型起動用スレッドプール定義ファイルを参照
					queueLength	デフォルトBean定義ファイルを参照
	4	jobPreProcessor	jp.terasoluna.fw.batch.restart.RestartPreProcessor	リスタート時のジョブ前処理の定義。	jobRestartTableHandler	項番21を参照
					preProcessor	項番5を参照
	5	restartJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	supportProcessor	項番6を参照
					standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。
6	-	jp.terasoluna.fw.batch.restart.RestartPostProcessor	リスタート時のジョブ後処理の定義。	jobRestartTableHandler	項番21を参照	
				postProcessor	項番8を参照	
7	jobPostProcessor	jp.terasoluna.fw.batch.restart.RestartPostProcessor	リスタート時のジョブ後処理の定義。	throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照	
				supportProcessor	項番9を参照	
8	delegatedJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ後処理の定義。	supportProcessor	項番9を参照	
				standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	
9	-	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	キュー処理クラスの定義。	name	初期値 [chunkTransactionQueueProcessor]	
				worker	項番11を参照	
ワーカ	11	chunkTransactionWorker	jp.terasoluna.fw.batch.spring.support.transactional.TransactionalWorker	チャック処理クラスの定義。	transactionManager	データアクセスBean定義ファイルを参照
					useSavepoint	デフォルトBean定義ファイルを参照
	12	restartPointUpdater	jp.terasoluna.fw.batch.restart.RestartPointUpdater	リスタートポイント登録用ワーククラスの定義。	jobWorker	項番12を参照
					jobRestartTableHandler	項番14を参照
	13	restartPointCloser	jp.terasoluna.fw.batch.restart.RestartPointCloser	リスタートポイント削除用ワーククラスの定義。	throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
					jobWorker	項番14を参照
	14	jobWorker	jp.terasoluna.fw.batch.core.JobWorker	チャック処理クラスの定義。	supportLogicList	デフォルトBean定義ファイルを参照
					sqlKey	ジョブBean定義ファイルを参照 リスタート情報クリア用SQL 初期設定値は [jobRestart.DELETE_JOB_RESTART_CLEAR] である。
	15	bllogicExecutor	jp.terasoluna.fw.batch.core.StandardBllogicExecutor	ビジネスロジック実行クラスの定義。	throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
					batchUpdateProcessor	項番16を参照
16	batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	exceptionHandlerMap	デフォルトBean定義ファイルを参照	
				batchUpdateResultHandler	デフォルトBean定義ファイルを参照	
17	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	defaultExceptionHandler	デフォルトBean定義ファイルを参照
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
18	-	JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	bllogic	ジョブBean定義ファイルを参照
					bllogicResultHandler	デフォルトBean定義ファイルを参照
19	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	exceptionHandlerMap	デフォルトBean定義ファイルを参照
					defaultExceptionHandler	デフォルトBean定義ファイルを参照
20	ジョブリスタート管理テーブルハンドラ	JobRestartInfoFactory	jp.terasoluna.fw.batch.restart.StandardJobRestartInfoFactory	ジョブリスタート情報用パラメータクラスのBean定義。	updateDAO	データアクセスBean定義ファイルを参照
					jobRestartInfoFactory	データアクセスBean定義ファイルを参照
21	ジョブリスタート情報用パラメータクラス	-	-	-	partitionNo	デフォルトBean定義ファイルを参照
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照

チャンク別トランザクションモデル (ブレイク)

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ				import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1	マネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	ブレースホルダの定義 ジョブ実行マネージャの定義。	name collector workQueueFactory preProcessor postProcessor	初期値 [chunkTransactionJobManager] ジョブBean定義ファイルを参照 番号2を参照 番号3を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common対応作業キュー生成クラスの定義。	workerExecutorsService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照
3		jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	supportProcessor	番号7を参照 番号4を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
5		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ後処理の定義。	supportProcessor	番号6を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
7	キュープロセッサ	chunkTransactionQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	キュー処理クラスの定義。	name worker	初期値 [chunkTransactionQueueProcessor] 番号8を参照
8	ワーカ	chunkTransactionWorker	TransactionWorker	トランザクション制御有りのワーカ定義。	transactionManager useSavepoint jobWorker	データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照 番号9を参照
9		controlBreakWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	logicExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler	番号10を参照 番号13を参照 デフォルトBean定義ファイルを参照 フレームワークBean定義ファイルを参照 デフォルトBean定義ファイルを参照
10		controlBreakLogicExecutor	jp.terasoluna.fw.batch.controlbreak.ControlBreakLogicExecutor	コントロールブレイク用ビジネスロジック実行クラスの定義。	controlBreakDef	番号14を参照
11		controlBreakProcessor	jp.terasoluna.fw.batch.controlbreak.ControlBreakProcessor	コントロールブレイク処理実行クラスの定義。	logicResultHandler exceptionHandlerMap defaultJobExceptionHandler throwableHandler	デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照。
12		bloticExecutor	jp.terasoluna.fw.batch.core.StandardLogicExecutor	ビジネスロジック実行クラスの定義。	logic logicResultHandler exceptionHandlerMap defaultJobExceptionHandler throwableHandler	ジョブBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照。
13		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	updateDAO	データアクセスBean定義ファイルを参照
14		コントロールブレイク	controlBreakDef	jp.terasoluna.fw.batch.controlbreak.ControlBreakDef	controlBreakDefItem chunkControlBreakDefItem transChunkControlBreakDefItem	ジョブBean定義ファイルを参照 ジョブBean定義ファイルを参照 ジョブBean定義ファイルを参照
12		ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	mbeanRegister transactionManager useSavepoint partitionNo defaultJobExitCodeMap	フレームワークBean定義ファイルを参照 データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [-1] デフォルトBean定義ファイルを参照
13			JobStatus	jp.terasoluna.fw.batch.core.JobStatus	partitionNo transactionManager useSavepoint partitionNo defaultJobExitCodeMap	初期値 [-1] データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [-1] デフォルトBean定義ファイルを参照
14		スレッドプールサイズ	threadSize	java.lang.Integer	constructor-arg	初期値 [1]

非トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1	マネージャ定義	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	ジョブ実行マネージャの定義。	name collector workQueueFactory preProcessor postProcessor	初期値 [noTransactionJobManager] ジョブBean定義ファイルを参照 項番2を参照 項番3を参照 項番4を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common)対応作業キュー生成クラスの定義。	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同同期型起動用スレッドプール定義ファイルを参照
3		jobPreProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	queueLength queueProcessor	デフォルトBean定義ファイルを参照 項番5を参照
4		jobPostProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
5		キュープロセッサ	noTransactionQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	キュー処理クラスの定義。	name worker
6	ワーカー	noTransactionWorker	baseWorker (jp.terasoluna.fw.batch.core.JobWorker)	チャンク処理クラスの定義。	bllogicExecutor exceptionHandlerMap defaultJobExceptionHandler batchUpdateProcessor batchUpdateResultHandler	項番7を参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 項番8を参照 デフォルトBean定義ファイルを参照
7		bllogicExecutor	jp.terasoluna.fw.batch.core.StandardBllogicExecutor	ビジネスロジック実行クラスの定義。	bllogic bllogicResultHandler exceptionHandlerMap defaultJobExceptionHandler throwableHandler	ジョブBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照
8		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	updateDAO	データアクセスBean定義ファイルを参照
9		ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister transactionManager useSavepoint partitionNo defaultJobExitCodeMap
10	ジョブの処理状況クラス	JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo transactionManager useSavepoint partitionNo defaultJobExitCodeMap	初期値 [-1] データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [-1] デフォルトBean定義ファイルを参照
11		スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	constructor-arg

単一トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ					
1	マネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	ブレースホルダの定義 ジョブ実行マネージャの定義。	import resource name collector workQueueFactory preProcessor postProcessor	初期値 [classpath:common/PlaceholderConfig.xml] 初期値 [singleTransactionJobManager] ジョブBean定義ファイルを参照 項目2を参照 項目3を参照 項目3を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応作業キュー生成クラスの定義。	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照
3		jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	supportProcessor	項目7を参照 項目4を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
5		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りのジョブ後処理の定義。	supportProcessor	項目6を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
7	キュープロセッサ	singleTransactionQueueProcessor	jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor	キュー処理クラスの定義。	name worker transactionManager preProcessor postProcessor throwableHandler	初期値 [singleTransactionQueueProcessor] 項目10を参照 データアクセスBean定義ファイルを参照 項目8を参照 項目9を参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
8		firstchunkPreProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	先頭チャンク前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
9		lastchunkPostProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	最終チャンク後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
10	ワーカー	singleTransactionWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	bllogicExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler throwableHandler	項目11を参照 項目12を参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
11		bllogicExecutor	jp.terasoluna.fw.batch.core.StandardBllogicExecutor	ビジネスロジック実行クラスの定義。	bllogic bllogicResultHandler exceptionHandlerMap defaultJobExceptionHandler throwableHandler	ジョブBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 デフォルトBean定義ファイルを参照 ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
12		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理クラスの定義。	updateDAO	データアクセスBean定義ファイルを参照
13	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	mbeanRegister transactionManager useSavepoint partitionNo defaultJobExitCodeMap	フレームワークBean定義ファイルを参照 データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [-1] デフォルトBean定義ファイルを参照
14		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo transactionManager useSavepoint partitionNo defaultJobExitCodeMap	初期値 [-1] データアクセスBean定義ファイルを参照 デフォルトBean定義ファイルを参照 初期値 [-1] デフォルトBean定義ファイルを参照
15	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	constructor-arg	初期値 [1]

ジョブ分割チャンク別トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1		jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャ定義。	name	初期値 [partitionChunkTransactionJobManager]
			collector	ジョブBean定義ファイルを参照	workQueueFactory	項番2を参照
2		-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応親ジョブ用作業キュー作成クラスの定義。	preProcessor	項番3を参照
			postProcessor	項番3を参照	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照
3	親ジョブマネージャ	parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りの親ジョブ前処理の定義。	queueLength	デフォルトBean定義ファイルを参照
			queueProcessor	項番7を参照	multiplicity	ジョブBean定義ファイルを参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ前処理の定義。	supportProcessor	項番4を参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
5		parentJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りの親ジョブ後処理の定義。	supportProcessor	項番6を参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ後処理の定義。	supportProcessor	項番6を参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
7	親ジョブキュープロセッサ	partitionChunkTransactionQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	name	初期値 [partitionChunkTransactionQueueProcessor]
8		childJobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	子ジョブ実行用マネージャ定義。	worker	項番8を参照
			collector	ジョブBean定義ファイルを参照	workQueueFactory	項番9を参照
9	子ジョブマネージャ	-	jp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応子ジョブ用作業キュー作成クラスの定義。	preProcessor	項番10を参照
			postProcessor	項番12を参照	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照
10	子ジョブマネージャ	jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りの子ジョブ前処理の定義。	queueLength	デフォルトBean定義ファイルを参照
			queueProcessor	項番14を参照	supportProcessor	項番11を参照
11		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
12	子ジョブマネージャ	jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション制御有りの子ジョブ後処理の定義。	supportProcessor	項番13を参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
13		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ後処理の定義。	supportProcessor	項番13を参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
14	子ジョブキュープロセッサ	partitionChunkTransactionChildQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	子ジョブ用キュー処理クラスの定義。	name	初期値 [partitionChunkTransactionChildQueueProcessor]
15		transactionWorker	jp.terasoluna.fw.batch.spring.support.transactionalTransactionWorker	トランザクション制御有りのワーカー定義。	worker	項番15を参照
			transactionManager	データアクセスBean定義ファイルを参照	useSavepoint	デフォルトBean定義ファイルを参照
16	ワーカー	chunkWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	jobWorker	ジョブBean定義ファイルを参照
			logicExecutor	項番18を参照	batchUpdateProcessor	項番17を参照
17		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理クラスの定義。	exceptionHandlerMap	デフォルトBean定義ファイルを参照
			batchUpdateResultHandler	デフォルトBean定義ファイルを参照	defaultExceptionHandler	デフォルトBean定義ファイルを参照
18		bllogicExecutor	jp.terasoluna.fw.batch.core.StandardBllogicExecutor	ビジネスロジック実行クラスの定義。	throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。
			updateDAO	データアクセスBean定義ファイルを参照		
19	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	bllogic	ジョブBean定義ファイルを参照
			bllogicResultHandler	デフォルトBean定義ファイルを参照	exceptionHandlerMap	デフォルトBean定義ファイルを参照
20		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	defaultExceptionHandler	デフォルトBean定義ファイルを参照
			throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照。	transactionManager	データアクセスBean定義ファイルを参照
21	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	useSavepoint	デフォルトBean定義ファイルを参照
			partitionNo	初期値『-1』	partitionNo	初期値『-1』
20		-			defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					multiplicity	ジョブBean定義ファイルを参照

ジョブ分割非トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1		jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャの定義。	name	初期値 [partitionNoTransactionJobManager]
			collector	ジョブBean定義ファイルを参照		
2	親ジョブマネージャ	-	kp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応親ジョブ用作業キュー作成クラスの定義。	workQueueFactory	項目2を参照
			preProcessor	項目3を参照		
3		parentJobPreProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ前処理の定義。	postProcessor	項目4を参照
			workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照		
4		parentJobPostProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ後処理の定義。	queueProcessor	項目5を参照
			queueLength	デフォルトBean定義ファイルを参照		
5	親ジョブキュープロセッサ	partitionNoTransactionQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	multiplicity	ジョブBean定義ファイルを参照
			supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照		
6	子ジョブマネージャ	childJobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	子ジョブ実行用マネージャの定義。	name	初期値 [childJobManager]
			collector	ジョブBean定義ファイルを参照		
7		-	kp.terasoluna.fw.batch.common.WorkManagerTaskWorkQueueFactory	common対応子ジョブ用作業キュー作成クラスの定義。	workQueueFactory	項目7を参照
			preProcessor	項目8を参照		
8		jobPreProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	postProcessor	項目9を参照
			workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照		
9		jobPostProcessor	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ後処理の定義。	queueLength	デフォルトBean定義ファイルを参照
			queueProcessor	デフォルトBean定義ファイルを参照		
10	子ジョブキュープロセッサ	partitionNoTransactionChildQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	子ジョブ用キュー処理クラスの定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
			name	初期値 [partitionNoTransactionChildQueueProcessor]		
11	ワーカ	noTransactionWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	worker	項目11を参照
			logicExecutor	項目12を参照		
12		blogicExecutor	jp.terasoluna.fw.batch.core.StandardLogicExecutor	ビジネスロジック実行クラスの定義。	exceptionHandlerMap	デフォルトBean定義ファイルを参照
			defaultJobExceptionHandler	デフォルトBean定義ファイルを参照		
13	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	batchUpdateProcessor	項目13を参照
			batchUpdateResultHandler	フレームワークBean定義ファイルを参照		
14		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照
			updateDAO	データベースBean定義ファイルを参照		
15	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	logicResultHandler	デフォルトBean定義ファイルを参照
			exceptionHandlerMap	デフォルトBean定義ファイルを参照		
16		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	defaultJobExceptionHandler	デフォルトBean定義ファイルを参照
			throwableHandler	ThrowableハンドラのBean定義。 初期設定値は [throwableHandler] である。 デフォルトBean定義ファイルを参照		

ジョブ分割単一トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1	jobManager		baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャの定義。	name	初期値
					collector	ジョブBean定義ファイル参照
2	-		jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common対応親ジョブ用作業キュー作成クラスの定義。	workQueueFactory	項目2を参照
					preProcessor	項目3を参照
3	parentJobPreProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの親ジョブ前処理の定義。	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照
					queueProcessor	非同同期型起動用スレッドプール定義ファイルを参照
4	-		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ前処理の定義。	queueLength	デフォルトBean定義ファイルを参照
					supportProcessor	ジョブBean定義ファイルを参照
5	parentJobPostProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの親ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照
					supportProcessor	ジョブBean定義ファイルを参照
6	-		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ後処理の定義。	supportProcessor	項目6を参照
					supportLogicList	デフォルトBean定義ファイルを参照
7	partitionSingleTransactionQueueProcessor		baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	name	初期値
					worker	項目9を参照
8	childJobManager		baseManager (jp.terasoluna.fw.batch.core.JobManager)	子ジョブ実行用マネージャの定義。	name	初期値 [childJobManager]
					collector	ジョブBean定義ファイルを参照
9	-		jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common対応子ジョブ用作業キュー作成クラスの定義。	workQueueFactory	項目9を参照
					preProcessor	項目10を参照
10	jobPreProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの子ジョブ前処理の定義。	postProcessor	項目12を参照
					workerExecutorService	同期型起動用スレッドプール定義ファイルを参照
11	-		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	queueProcessor	非同同期型起動用スレッドプール定義ファイルを参照
					queueLength	項目14を参照
12	jobPostProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの子ジョブ後処理の定義。	queueLength	デフォルトBean定義ファイルを参照
					supportProcessor	ジョブBean定義ファイルを参照
13	-		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ後処理の定義。	supportProcessor	項目13を参照
					supportLogicList	デフォルトBean定義ファイルを参照
14	singleTransactionChildQueueProcessor		baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	子ジョブ用キュー処理クラスの定義。	name	初期値
					worker	項目11を参照
15	firstChunkPreProcessor		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	先頭チャンク前処理の定義。	preProcessor	項目15を参照
					postProcessor	項目16を参照
16	lastChunkPostProcessor		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	最終チャンク後処理の定義。	throwableHandler	ThrowableハンドラのBean定義。初期設定値は『throwableHandler』である。
					supportLogicList	デフォルトBean定義ファイルを参照
17	singleTransactionWorker		jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	batchUpdateProcessor	項目18を参照
					exceptionHandlerMap	デフォルトBean定義ファイルを参照
18	batchUpdateProcessor		jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理クラスの定義。	defaultJobExceptionHandler	デフォルトBean定義ファイルを参照
					batchUpdateResultHandler	デフォルトBean定義ファイルを参照
19	bllogicExecutor		jp.terasoluna.fw.batch.core.StandardBlLogicExecutor	ビジネスロジック実行クラスの定義。	throwableHandler	ThrowableハンドラのBean定義。初期設定値は『throwableHandler』である。
					updateDAO	データベースBean定義ファイルを参照
20	ジョブの処理状況クラス	MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	bllogic	ジョブBean定義ファイルを参照
					bllogicResultHandler	ジョブBean定義ファイルを参照
21	JobStatus		jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	exceptionHandlerMap	デフォルトBean定義ファイルを参照
					defaultJobExceptionHandler	デフォルトBean定義ファイルを参照
22	スレッドプールサイズ	threadSize	java.lang.Integer	スレッドプールのサイズ	throwableHandler	ThrowableハンドラのBean定義。初期設定値は『throwableHandler』である。
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照

ジョブ分割逐次単一トランザクションモデル

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
0	ブレースホルダ			ブレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1	jobManager		baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャの定義。	name	初期値
					collector	ジョブBean定義ファイル参照
2	-		jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common対応親ジョブ用作業キュー作成クラスの定義。	workQueueFactory	項目2を参照
					preProcessor	項目3を参照
					workerExecutorService	同期型起動用スレッドプール定義ファイルを参照
					postProcessor	非同期型起動用スレッドプール定義ファイルを参照
					queueProcessor	項目7を参照
					queueLength	デフォルトBean定義ファイルを参照
3	parentJobPreProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの親ジョブ前処理の定義。	supportProcessor	ジョブBean定義ファイルを参照
						項目4を参照
4	-	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
5	parentJobPostProcessor	-	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの親ジョブ後処理の定義。	supportProcessor	項目6を参照
6	-	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
7	親ジョブキュープロセッサ		sequentialSingleTransactionQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	親ジョブ用キュー処理クラスの定義。	name	初期値 [sequentialSingleTransactionQueueProcessor]
8	childJobManager		baseManager (jp.terasoluna.fw.batch.core.JobManager)	子ジョブ実行用マネージャの定義。	worker	項目9を参照
					name	初期値 [childJobManager]
9	-		jp.terasoluna.fw.batch.common.WorkManagerTaskWorkerQueueFactory	common対応子ジョブ用作業キュー作成クラスの定義。	collector	ジョブBean定義ファイルを参照
					workQueueFactory	項目9を参照
					preProcessor	項目10を参照
					postProcessor	項目12を参照
					workerExecutorService	同期型起動用スレッドプール定義ファイルを参照
					queueProcessor	非同期型起動用スレッドプール定義ファイルを参照
10	jobPreProcessor		transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの子ジョブ前処理の定義。	supportProcessor	ジョブBean定義ファイルを参照
						項目11を参照
11	-	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
12	jobPostProcessor	-	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transactionalSupportProcessor)	トランザクション管理有りの子ジョブ後処理の定義。	supportProcessor	項目13を参照
13	-	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
14	子ジョブキュープロセッサ		baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	子ジョブ用キュー処理クラスの定義。	name	初期値 [singleTransactionChildQueueProcessor]
					worker	項目11を参照
15	firstChunkPreProcessor		standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	先頭チャンク前処理の定義。	supportLogicList	初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照
						ジョブBean定義ファイルを参照
16	lastChunkPostProcessor	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	最終チャンク後処理の定義。	supportLogicList	デフォルトBean定義ファイルを参照 ジョブBean定義ファイルを参照
17	singleTransactionWorker		jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	batchExecutor	項目19を参照
					batchUpdateProcessor	項目18を参照
					exceptionHandlerMap	デフォルトBean定義ファイルを参照
					defaultJobExceptionHandler	デフォルトBean定義ファイルを参照
					batchUpdateResultHandler	デフォルトBean定義ファイルを参照
					throwableHandler	ThrowableハンドラのBean定義。 初期設定値は『throwableHandler』である。 デフォルトBean定義ファイルを参照
18	batchUpdateProcessor	-	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理クラスの定義。	updateDAO	データベースBean定義ファイルを参照
19	batchExecutor		jp.terasoluna.fw.batch.core.StandardBatchLogicExecutor	ビジネスロジック実行クラスの定義。	batchLogic	ジョブBean定義ファイルを参照
					batchResultHandler	ジョブBean定義ファイルを参照
20	monitorableJobStatus		jp.terasoluna.fw.batch.monitor.MonitorableJobStatus	監視用ジョブの処理状況用Bean定義。	beanRegister	フレームワークBean定義ファイルを参照
					transactionManager	データベースBean定義ファイルを参照
21	jobStatus		jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	useSavepoint	デフォルトBean定義ファイルを参照
					partitionNo	初期値 [-1]
22	分割ジョブの多重起動数	multiplicity	java.lang.Integer	分割ジョブの多重起動数	defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					constructor-arg	初期値 [1]

ファイルアクセス

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要	
1	CSVファイルアクセス用 (入力) DAO	csvFileQueryDAO	jp.terasoluna.fw.file.dao.standard.CSVFileQueryDAO / fileQueryDAO	CSVファイルアクセス用 (入力) DAO	-	-	
2	固定長ファイルアクセス用 (入力) DAO	fixedFileQueryDAO	jp.terasoluna.fw.file.dao.standard.FixedFileQueryDAO / fileQueryDAO	固定長ファイルアクセス用 (入力) DAO	-	-	
3	可変長ファイルアクセス用 (入力) DAO	variableFileQueryDAO	jp.terasoluna.fw.file.dao.standard.VariableFileQueryDAO / fileQueryDAO	可変長ファイルアクセス用 (入力) DAO	-	-	
4	その他ファイルアクセス用 (入力) DAO	plainFileQueryDAO	jp.terasoluna.fw.file.dao.standard.PlainFileQueryDAO / fileQueryDAO	その他ファイルアクセス用 (入力) DAO	-	-	
5	ファイルアクセス用 (入力) DAO 共通定義	fileQueryDAO	jp.terasoluna.fw.file.dao.standard.AbstractFileQueryDAO	ファイルアクセス用 (入力) DAO 共通定義	textSetterMap	内容については項番6~9を参照	
6		-	key:java.lang.String jp.terasoluna.fw.file.dao.standard.TextSetString	カラムフォーマットクラスの定義。(String型)	validator	項番10を参照	
7		-	key:java.util.Date jp.terasoluna.fw.file.dao.standard.TextSetDate	カラムフォーマットクラスの定義。(Date型)	-	-	
8		-	key:java.math.BigDecimal jp.terasoluna.fw.file.dao.standard.TextSetDecimal	カラムフォーマットクラスの定義。(BigDecimal型)	-	-	
9		-	key:int jp.terasoluna.fw.file.dao.standard.TextSetInt	カラムフォーマットクラスの定義。(int型)	-	-	
10		-	org.springframework.validation.commons.DefaultBeanValidator	バリデータのサンプルBean定義。	validatorFactory	項番11を参照	
11		validatorFactory	jp.terasoluna.fw.validation.springmodules.DefaultValidatorFactoryEx	ファイル入力チェック (バリデータファクトリの記述例)	validationConfigLocations	-	
12		CSVファイルアクセス用 (出力) DAO	csvFileUpdateDAO	jp.terasoluna.fw.file.dao.standard.CSVFileUpdateDAO / fileUpdateDAO	CSVファイルアクセス用 (出力) DAO	-	-
13		固定長ファイルアクセス用 (出力) DAO	fixedFileUpdateDAO	jp.terasoluna.fw.file.dao.standard.FixedFileUpdateDAO / fileUpdateDAO	固定長ファイルアクセス用 (出力) DAO	-	-
14		可変長ファイルアクセス用 (出力) DAO	variableFileUpdateDAO	jp.terasoluna.fw.file.dao.standard.VariableFileUpdateDAO / fileUpdateDAO	可変長ファイルアクセス用 (出力) DAO	-	-
15	その他ファイルアクセス用 (出力) DAO	plainFileUpdateDAO	jp.terasoluna.fw.file.dao.standard.PlainFileUpdateDAO / fileUpdateDAO	その他ファイルアクセス用 (出力) DAO	-	-	
16	ファイルアクセス用 (出力) DAO 共通定義	fileUpdateDAO	jp.terasoluna.fw.file.dao.standard.AbstractFileUpdateDAO	ファイルアクセス用 (出力) DAO 共通定義	textGetterMap	内容については項番17~20を参照	
17		-	key:java.lang.String jp.terasoluna.fw.file.dao.standard.TextGetString	ファイル書き込み用カラムフォーマットクラスの定義。(String型)	-	-	
18		-	key:java.util.Date jp.terasoluna.fw.file.dao.standard.TextGetDate	ファイル書き込み用カラムフォーマットクラスの定義。(Date型)	-	-	
19		-	key:java.math.BigDecimal jp.terasoluna.fw.file.dao.standard.TextGetDecimal	ファイル書き込み用カラムフォーマットクラスの定義。(BigDecimal型)	-	-	
20		-	key:int jp.terasoluna.fw.file.dao.standard.TextGetInt	ファイル書き込み用カラムフォーマットクラスの定義。(int型)	-	-	