



TERASOLUNA Batch Framework for Java

設定ファイル説明書

第 2.0.0.2 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確信や信頼性の保証、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 ver2.0.0.2 に対応しています。

Bean定義一覧

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	非同期型ジョブ起動機能バッチデーモン(メッセージ送信用)定義ファイル	AsyncBatchDaemonForSendMessageBean.xml
11	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
12	ワーク用ワークマネージャ定義ファイル	WorkManagerTaskContext-batch.xml
13	プレースホルダ設定定義ファイル	PlaceholderConfig.xml
14	チャンク別トランザクションモデル	ChunkTransactionBean.xml
15	チャンク別トランザクションモデル(リスタート)	ChunkTransactionForRestartBean.xml
16	チャンク別トランザクションモデル(ブレイク)	ChunkTransactionForControlBreakBean.xml
17	非トランザクションモデル	NoTransactionBean.xml
18	単一トランザクションモデル	SingleTransactionBean.xml
19	ジョブ分割チャンク別トランザクションモデル	PartitionChunkTransactionBean.xml
20	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
21	ジョブ分割非トランザクションモデル	PartitionNoTransactionBean.xml
22	ジョブ分割単一トランザクションモデル	PartitionSingleTransactionBean.xml
23	ジョブ分割逐次単一トランザクションモデル	SequentialSingleTransactionBean.xml
24	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		messageAcceptor	jp.terasoluna.fw.batch.springsupport.messages.MessageAcceptorImpl	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 collectedDataHandlerFactory	使用するQueryDao。 初期設定値は『queryDAO』である。 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。
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		stringArrayCollector	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 collectedDataHandlerFactory	使用するQueryDao。 初期設定値は『queryDAO』である。 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。
			jp.terasoluna.fw.batch.standard.ChunkerFactory	チャンカー作成クラスの定義。	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		validationStringArrayCollector	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 collectedDataHandlerFactory	使用するQueryDao。 初期設定値は『queryDAO』である。 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			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用	collectedDataHandlerFactory validator	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。 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 exceptionHandlerMap defaultJobExceptionHandler	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。 例外ハンドラのBean定義 初期設定値は『exceptionHandlerMap』である。 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。
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 defaultJobExceptionHandler exceptionHandlerMap	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。 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。 例外ハンドラを格納したMapのBean定義。 初期設定値は『deExceptionHandlerMap』である。
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の要素をパラメータとしてワーカー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。	-	-
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 defaultJobExceptionHandler exceptionHandlerMap	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。 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。

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	分割ジョブの単トランザクションモデルのBean定義ファイル	
9					SingleTransactionBean.xml	単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	観ジョブ前処理	parentJobPreLogicList	org.springframework.beans.factory.config.ListFactoryBean	分割ジョブ実行時の観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	観ジョブ後処理	parentJobPostLogicList	org.springframework.beans.factory.config.ListFactoryBean	分割ジョブ実行時の観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 『ibatisDbPartitionKeyCollector』を選択した場合は『sql』	『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	blogic	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	sqlMapConfigFileName	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	controlBreakDefItem	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			jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem	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					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	chunkControlBreakDefItem	jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem	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					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	transControlBreakDefItem	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			jp.terasoluna.fw.batch.controlbreak.ControlBreakDefItem	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					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のパス 初期設定値は『batchapps/BE-03/EndFile』である。
2		scheduledTask	org.springframework.scheduling.timer.ScheduledTimerTask	スケジュールタスク用Bean定義。 CommonJ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。 スケジュール起動対象。 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。 CommonJ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。 スケジュールタスク用設定。 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:1 Key:SUSPENDING value:2
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『CollectorExceptionHandler:jp.terasoluna.batch.sample.bo01.CollectorExceptionHandlerImpl JobExceptionHandler: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	JobExceptionHandler用デフォルト例外ハンドラのBean定義。	なし	
22	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	
23		TransactionalBLogicResultHandler	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalBLogicResultHandler	チャング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	
24		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	
25		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	
26		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	
27	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。
28		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. dbc. p. 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）
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。
					defaultAutoCommit	オートコミットの有無（デフォルトはfalse）
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。
					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	ストアドプロシージャ用のDAOのBean定義。	sqlMapClient	使用するSQLMapClientの設定。 初期設定値は『sqlMapClient』である。

非同同期型ジョブ起動機能パッチデモン

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	非同同期バッチ起動用マネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	非同同期バッチ起動用マネージャの定義。	name	初期値『AsyncBatchDaemonManager』
					collector	項番16を参照
					workQueueFactory	項番2を参照
					preProcessor	項番3を参照
					postProcessor	項番5を参照
2		-	jp.terasoluna.fw.batch.standard.StandardWorkQueueFactory	非同同期バッチ起動用作業キュー作成クラスの定義。	workerExecutorService	フレームワークBean定義ファイルを参照
					queueProcessor	項番8を参照
					queueLength	初期値『10』
					multiplicity	項番29を参照
3	非同同期バッチ起動用マネージャ	parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.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.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	初期値『asyncBatchDaemonQueueProcessor』
					worker	項番9を参照
9	バッチ起動用ワーク	childJobManager	jp.terasoluna.fw.batch.springsupport.init.JobExecutor	ジョブ起動用クラスの定義。	constructor-arg	Bean定義ファイル 初期値 『/common/FrameworkBean.xml./common/dataAccessContext-batch.xml./common/ThreadPoolContext-AsyncBatch.xml』
					async	非同同期起動有無フラグ 初期値『true』
					preJobProcessor	項番10を参照
					postJobProcessor	項番13を参照
10		preJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.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	非同同期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	コレクター	postJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.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.AsyncJobPreLogic	非同同期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		batchDaemonJobCollector	jp.terasoluna.fw.batch.init.JobRequestInfoCollector	非同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	項番17を参照
					jobControlTableHandler	フレームワークBean定義ファイルを参照
					intervalSeconds	ジョブ管理テーブルの監視周期 (秒) 初期値『10』
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。	-	-
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。
20					useSavepoint	データアクセスBean定義ファイルを参照。
21					partitionNo	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。
22	ジョブの処理状況クラス	JobStatus	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalJobStatus	ジョブの処理状況用Bean定義。	transactionManager	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。
23					useSavepoint	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。
24					partitionNo	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。
25		useMonitorable	org.springframework.beans.factory.config.FieldRetrievingFactoryBean	ジョブ監視有無のBean定義。	staticField	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。
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		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					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	スレッド数の設定。 初期設定値は『20』である。

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	workerExecuterService	jp.terasoluna.fw.batch.springsupport.standard.FixedThreadPoolWorkerExecuterService	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	スレッド数の設定。 初期設定値は『threadSize』である。

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	初期値『AsyncBatchDaemonManager』
2		-	jp.terasoluna.fw.batch.commonj.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)。	collector workQueueFactory preProcessor postProcessor	項番16を参照 項番2を参照 項番3を参照 項番5を参照
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.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	項番8を参照 項番29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.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	初期値『asyncBatchDaemonQueueProcessor』
9	バッチ起動用ワーク	childJobManager	jp.terasoluna.fw.batch.commonj.init.JobExecutor	ジョブ起動用クラスの定義。	worker constructor-arg	項番9を参照 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』
10			transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalSupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	async preJobProcessor postJobProcessor supportProcessor	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 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.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 intervalSeconds refreshCount	項番17を参照 フレームワークBean定義ファイルを参照 ジョブ管理テーブルの監視周期 (秒) 初期値『10』 初期値『20』
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。	-	-
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		ジョブの処理状況クラス			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.springsupport.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		ジョブの処理状況クラス			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	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		multiplicity	java.lang.Integer	多重度	constructor-arg	初期値『3』

非同期型ジョブ起動機能パッチデーモン(メッセージ送信用)

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	非同期パッチ起動用マネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	非同期パッチ起動用マネージャの定義。	name	初期値『AsyncBatchDaemonManager』
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	非同期パッチ起動用作業キュー作成クラスの定義 (CommonJ用)。	collector	項番16を参照
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	workQueueFactory	項番2を参照
4		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	preProcessor	項番3を参照
5		parentJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	postProcessor	項番5を参照
6		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	workerExecutorService	フレームワークBean定義ファイルを参照
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。	queueProcessor	項番8を参照
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。	queueLength	初期値『10』
9	パッチ起動用ワーク	childJobManager	jp.terasoluna.fw.batch.commonj.init.JobExecutor	ジョブ起動用クラスの定義。	multiplicity	項番29を参照
10		preJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りのジョブ前処理の定義。	name	初期値『asyncBatchDaemonQueueProcessor』
11		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ前処理の定義。	worker	項番9を参照
12		-	jp.terasoluna.fw.batch.jobmessage.SendMessagePreLogic	メッセージ送信用ジョブ前処理の定義。	constructor-arg	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』
13		postJobProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りのジョブ後処理の定義。	async	非同期起動有無フラグ 初期値『true』
14		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	ジョブ後処理の定義。	preJobProcessor	項番10を参照
15		-	jp.terasoluna.fw.batch.jobmessage.SendMessagePostLogic	メッセージ送信用ジョブ後処理の定義。	postJobProcessor	項番13を参照
16	コレクター	batchDaemonJobCollector	jp.terasoluna.fw.batch.jobmessage.SendMessageResultInfoCollector	メッセージ送信ジョブ起動時のジョブ結果情報取得用クラスの定義。	supportProcessor	項番11を参照
17		-	jp.terasoluna.fw.batch.jobmessage.JobMessageInfoHandlerFactory	メッセージ送信情報格納用ワークキュー生成用クラスの定義。	supportLogicList	項番12を参照
18		MonitorableJobStatus	jp.terasoluna.fw.batch.monitor.MonitorableTransactionalJobStatus	監視用ジョブの処理状況用Bean定義。	jobMessageTableHandler	フレームワークBean定義ファイルを参照
19	ジョブの処理状況クラス定義	-	-	-	jobMessageTableHandler	フレームワークBean定義ファイルを参照
20		-	-	-	intervalSeconds	ジョブ管理テーブルの監視周期 (秒) 初期値『10』
21		-	-	-	refreshCount	初期値『20』
22	ジョブの処理状況クラス	JobStatus	jp.terasoluna.fw.batch.springsupport.transaction.Transactionalsupport.transaction.TransactionalsupportJobStatus	ジョブの処理状況用Bean定義。	jobBeanPath	初期値『common/MessageSendJob.xml』
23		-	-	-	-	-
24		-	-	-	-	-
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	ジョブメッセージテーブル	jobMessageTableHandler	jp.terasoluna.fw.batch.init.JobMessageTableHandlerImpl	ジョブメッセージ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		multiplicity	java.lang.Integer	多重度	constructor-arg	初期値『3』

ジョブス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	jobStarter	jp.terasoluna.fw.batch.commonj.init.JobStarterImpl	ジョブ起動の定義	workManager	項番2を参照。 初期値は『manager』である。
					workListener	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	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	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。
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	参照系のDAOのBean定義。 初期値は『queryDAO』である。
					updateDAO	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	参照系のDAOのBean定義。 初期値は『updateDAO』である。
					transactionManager	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

項番	設定項目名称	Bean ID	Classes/parent名	概要	property	property概要
1	ワーク管理	workerExecutorService	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。
					executor	項番2を参照。 初期値は『taskExecutor』である。
					workListener	項番3を参照。 初期値は『workListener』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。
3		workListener	jp.terasoluna.fw.batch.commonj.listener.WorkQueueListener	ワークリスナーの定義	resourceRef	初期値は『true』である。
						-
4	監視	scheduledTask	org.springframework.scheduling.commonj.ScheduledTimerListener	スケジューラーの定義 Commonj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。	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		timerFactory	org.springframework.scheduling.commonj.TimerManagerFactoryBean	タイマーの定義 Commonj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。	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。

ブレースフォ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	ブレースホルダ	-	org.springframework.beans.factory.config.PropertyPlaceholderConfigurer	ジョブが使用するブレースホルダの設定定義。	locations	使用するブ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

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	jobManager	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
					workQueueFactory	項番2を参照
					preProcessor	項番3を参照
					postProcessor	項番5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
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応作業キュー生成クラスの定義。	queueLength	非同期型起動用スレッドプール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
3	マネージャ	jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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
5		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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
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	初期値 [chunkTransactionQueueProcessor]
					worker	項番8を参照
8	ワーカ	chunkTransactionWorker	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalWorker	トランザクション制御有りのワーカ定義。	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
					jobWorker	ジョブBean定義ファイルを参照
9		jobWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	bloticExecutor	項番10を参照
					batchUpdateProcessor	項番11を参照
					exceptionHandlerMap	デフォルトBean定義ファイルを参照
10		bloticExecutor	jp.terasoluna.fw.batch.core.StandardBloticExecutor	ビジネスロジック実行クラスの定義。	batchUpdateResultHandler	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
11		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	blotic	ジョブBean定義ファイルを参照
					bloticResultHandler	デフォルト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。	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		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo	初期値 [-1]
					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
					partitionNo	初期値 [-1]
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	defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					constructor-arg	初期値 [1]

チャン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』
1	マネージャ	jobManager	jp.terasoluna.fw.batch.restart.JobContextRestorer	リスタート実行用マネージャ定義。	jobRestartTableHandler jobManager	項番21を参照 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 collector workQueueFactory preProcessor postProcessor	初期値『chunkTransactionJobManager』 ジョブBean定義ファイルを参照 項番3参照 項番4を参照 項番7を参照
3		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応作業キュー生成クラスの定義。	workerExecutorService 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10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 preProcessor	項番21を参照 項番5を参照
5		delegatedJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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
6		-	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
7		jobPostProcessor	jp.terasoluna.fw.batch.restart.RestartPostProcessor	リスタート時のジョブ後処理の定義。	jobRestartTableHandler postProcessor	項番21を参照 項番8を参照
8		delegatedJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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
9		-	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
10	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11を参照
11	ワーカ	chunkTransactionWorker	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportTransactionWorker	チャンク処理クラスの定義。	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 項番12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 jobRestartTableHandler	項番13を参照 項番21を参照
13		jobWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	logicExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler	項番14を参照 項番15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
14		logicExecutor	jp.terasoluna.fw.batch.core.StandardBLogicExecutor	ビジネスロジック実行クラスの定義。	logic logicResultHandler exceptionHandlerMap defaultExceptionHandler	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
15		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
16	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
17		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
18		threadSize	java.lang.Integer	スレッドプールのサイズ	constructor-arg	初期値『1』
19	ジョブリスタート管理テーブルハンドラ	JobRestartTableHandler	jp.terasoluna.fw.batch.restart.JobRestartTableHandler	ジョブリスタート管理テーブルのハンドラクラス用Bean定義。	queryDAO updateDAO jobRestartInfoFactory	データアクセス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 項番22を参照
20	ジョブリスタート情報用パラメータクラス	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。	なし	なし

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	-	-	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	jobManager	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
					workQueueFactory	項番2を参照
		jobPreProcessor	jp.terasoluna.fw.batch.common.JobManagerTaskWorkQueueFactory	commonJ対応作業キュー生成クラスの定義。	preProcessor	項番3を参照
					postProcessor	項番5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
2		-	-	-	queueLength	デフォルトBean定義ファイルを参照
3	マネージャ	jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	queueProcessor	項番7を参照
					supportProcessor	項番4を参照
		jobPostProcessor	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
					supportProcessor	項番6を参照
4		-	-	-	-	-
5		-	-	-	-	-
6		-	-	-	-	-
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	初期値 [chunkTransactionQueueProcessor]
					worker	項番8を参照
8	ワーカ	chunkTransactionWorker	jp.terasoluna.fw.batch.springsupport.transaction.TransactionalWorker	トランザクション制御有りのワーカ定義。	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 ジョブBean定義ファイルを参照
		controlBreakWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	jobWorker	項番9を参照
					logicExecutor	項番10を参照
		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。	batchUpdateProcessor	項番13を参照
					exceptionHandlerMap	デフォルトBean定義ファイルを参照
		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。	batchUpdateResultHandler	フレームワーク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
		blogicExecutor	jp.terasoluna.fw.batch.core.StandardBlogicExecutor	ビジネスロジック実行クラスの定義。	blogicExecutor	項番12を参照
					controlBreakProcessor	項番11を参照
		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	controlBreakDef	デフォルトBean定義ファイルを参照
					blogicResultHandler	項番14を参照
9	ワーカ	controlBreakDef	jp.terasoluna.fw.batch.controlbreak.ControlBreakDef	コントロールブレイク情報クラスの定義。	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
10		-	-	-	-	-
11		-	-	-	-	-
12		-	-	-	-	-
13		-	-	-	-	-
14		-	-	-	-	-
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。	updateDAO	データアクセスBean定義ファイルを参照
		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	controlBreakDefItemList	ジョブBean定義ファイルを参照
13	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。	chunkControlBreakDefItem	ジョブBean定義ファイルを参照
					transChunkControlBreakDefItem	ジョブ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	mbeanRegister	フレームワーク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
					useSavepoint	デフォルトBean定義ファイルを参照
					partitionNo	初期値 [-1]
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					partitionNo	初期値 [-1]
					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
					partitionNo	初期値 [-1]
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					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	初期値『noTransactionJobManager』
					collector	ジョブBean定義ファイルを参照
					workQueueFactory	項番2参照
					preProcessor	項番3を参照
2	マネージャ定義	-	jp.terasoluna.fw.batch.common.JWorkManagerTaskWorkQueueFactory	commonj対応作業キュー生成クラスの定義。	postProcessor	項番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
					queueLength	デフォルトBean定義ファイルを参照
					queueProcessor	項番5を参照
3	マネージャ定義	jobPreProcessor	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
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	初期値『noTransactionQueueProcessor』
					worker	項番6を参照
6	ワーカ	noTransactionWorker	baseWorker (jp.terasoluna.fw.batch.core.JobWorker)	チャック処理クラスの定義。	blogicExecutor	項番7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
					batchUpdateProcessor	項番8を参照
7	ワーカ	blogicExecutor	jp.terasoluna.fw.batch.core.StandardBlogicExecutor	ビジネスロジック実行クラスの定義。	batchUpdateResultHandler	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。	blogic	ジョブ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。	blogicResultHandler	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
					defaultJobExceptionHandler	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
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。	mbeanRegister	フレームワーク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
					useSavepoint	デフォルトBean定義ファイルを参照
					partitionNo	初期値『-1』
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	defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					partitionNo	初期値『-1』
					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
					partitionNo	初期値『-1』
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
					constructor-arg	初期値『1』

単一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	プレースホルダ	-	-	プレースホルダの定義	import resource	初期値 『classpath:common/PlaceholderConfig.xml』
1	マネージャ	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『singleTransactionJobManager』 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5を参照
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj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
3		jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionaSupportProcessor)	トランザクション制御有りのジョブ前処理の定義。	queueLength queueProcessor supportProcessor	デフォルトBean定義ファイルを参照 項番7を参照 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.springsupport.transaction.TransactionaSupportProcessor)	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	singleTransactionQueueProcessor	jp.terasoluna.fw.batch.springsupport.transaction.TransactionaQueueProcessor	キュー処理クラスの定義。	name worker transactionManager preProcessor postProcessor	初期値『singleTransactionQueueProcessor』 項番10を参照 データアクセスBean定義ファイルを参照 項番8を参照 項番9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。	bloagicExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler	項番11を参照 項番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
11		bloagicExecutor	jp.terasoluna.fw.batch.core.StandardBLoagicExecutor	ビジネスロジック実行クラスの定義。	bloagic bloagicResultHandler exceptionHandlerMap defaultJobExceptionHandler	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
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	プレースホルダ	-	-	プレースホルダの定義	import resource	初期値 [classpath:common/PlaceholderConfig.xml]
1	親ジョブマネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャ定義。	name collector workQueueFactory preProcessor postProcessor workerExecutorService	初期値 ジョブBean定義ファイルを参照 項番2参照 項番3参照 項番5参照 同期型起動用スレッドプール定義ファイルを参照
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応親ジョブ用作業キュー作成クラスの定義。	queueLength queueProcessor multiplicity	デフォルトBean定義ファイルを参照 項番7を参照 ジョブBean定義ファイルを参照
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション制御有りの親ジョブ前処理の定義。	supportProcessor	項番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.transaction.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
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	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 worker	初期値 [partitionChunkTransactionQueueProcessor]
8	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定義。	name collector workQueueFactory preProcessor postProcessor workerExecutorService	初期値 初期値 初期値 ジョブBean定義ファイルを参照 項番9参照 項番10参照 項番12参照 同期型起動用スレッドプール定義ファイルを参照
9		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応子ジョブ用作業キュー作成クラスの定義。	queueLength queueProcessor	デフォルトBean定義ファイルを参照 項番14を参照
10		jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.TransactionalSupportProcessor)	トランザクション制御有りの子ジョブ前処理の定義。	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
12		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.spring.support.transaction.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
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	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 worker	初期値 [partitionChunkTransactionChildQueueProcessor]
15	ワーカー	transactionWorker	jp.terasoluna.fw.batch.spring.support.transaction.TransactionalWorker	トランザクション制御有りのワーカー定義。	transactionManager useSavepoint	初期値 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
16		chunkWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	jobWorker blockExecutor batchUpdateProcessor exceptionHandlerMap batchUpdateResultHandler defaultJobExceptionHandler	項番16を参照 項番18を参照 項番17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
17		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
18		blockExecutor	jp.terasoluna.fw.batch.core.StandardBlockExecutor	ビジネスロジック実行クラスの定義。	blockic blockicResultHandler exceptionHandlerMap defaultJobExceptionHandler	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
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。	mbeanRegister transactionManager useSavepoint	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
20		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	partitionNo defaultJobExitCodeMap	初期値 [-1] デフォルト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	partitionNo transactionManager useSavepoint partitionNo defaultJobExitCodeMap multiplicity	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 ジョブBean定義ファイルを参照

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)

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	親ジョブマネージャ	jobManager	jp.terasoluna.fw.batch.restart.JobContextRestorer	親ジョブリスタート実行用マネージャの定義。	jobRestartTableHandler	項番31を参照
2		delegatedJobManager	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』
3		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応親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
4		-	-	-	workQueueFactory	項番3を参照
5		-	-	-	preProcessor	項番4を参照
6		-	-	-	postProcessor	項番7を参照
7		-	-	-	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
8		-	-	-	queueLength	デフォルトBean定義ファイルを参照
9		-	-	-	multiplicity	ジョブBean定義ファイルを参照
10	親ジョブキュープロセッサ	parentJobPreProcessor	jp.terasoluna.fw.batch.restart.RestartPreProcessor	リスタート時の親ジョブ前処理の定義。	jobRestartTableHandler	項番31を参照
11		delegatedParentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りの親ジョブ前処理の定義。	preProcessor	項番5を参照
12		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ前処理の定義。	parentManager	初期値『true』
13		parentJobPostProcessor	jp.terasoluna.fw.batch.restart.RestartPostProcessor	リスタート時の親ジョブ後処理の定義。	supportProcessor	項番6を参照
14		delegatedParentJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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 ジョブBean定義ファイルを参照
15		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	親ジョブ後処理の定義。	jobRestartTableHandler	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		-	-	-	postProcessor	項番8を参照
17		-	-	-	supportProcessor	項番9を参照
18		-	-	-	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
19	子ジョブマネージャ	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』
20		childJobManager	jp.terasoluna.fw.batch.restart.JobContextRestorer	子ジョブリスタート実行用マネージャの定義。	worker	項番11を参照
21		delegatedChildJobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	子ジョブ実行用マネージャ定義。	jobRestartTableHandler	項番31を参照
22		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応子ジョブ用作業キュー作成クラスの定義。	jobManager	項番12を参照
23		jobPreProcessor	jp.terasoluna.fw.batch.restart.RestartPreProcessor	リスタート時の子ジョブ前処理の定義。	name	初期値『childJobManager』
24		delegatedJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りの子ジョブ前処理の定義。	collector	ジョブBean定義ファイルを参照
25		-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ前処理の定義。	workQueueFactory	項番13参照
26		jobPostProcessor	jp.terasoluna.fw.batch.restart.RestartPostProcessor	リスタート時の子ジョブ後処理の定義。	preProcessor	項番14を参照
27		delegatedJobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalsupportProcessor)	トランザクション管理有りの子ジョブ後処理の定義。	postProcessor	項番17を参照
28	子ジョブキュープロセッサ	-	standardSupportProcessor (jp.terasoluna.fw.batch.standard.StandardSupportProcessor)	子ジョブ後処理の定義。	workerExecutorService	同期型起動用スレッドプール定義ファイルを参照 非同期型起動用スレッドプール定義ファイルを参照
29		partitionChunkTransactionChildQueueProcessor	baseQueueProcessor (jp.terasoluna.fw.batch.standard.QueueProcessor)	子ジョブ用キュー処理クラスの定義。	queueLength	デフォルトBean定義ファイルを参照
30		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。	queueProcessor	項番20を参照
31		chunkWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	jobRestartTableHandler	項番31を参照
32		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	preProcessor	項番15を参照
33		blogicExecutor	jp.terasoluna.fw.batch.core.StandardBlogicExecutor	ビジネスロジック実行クラスの定義。	supportProcessor	項番16を参照
34		-	-	-	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
35		-	-	-	jobRestartTableHandler	デフォルトBean定義ファイルを参照
36		-	-	-	postProcessor	項番18を参照
37	ワーカ	transactionWorker	jp.terasoluna.fw.batch.springsupport.transaction.Transactionalsupport.transaction.TransactionalsupportWorker	チャンク処理クラスの定義。	supportProcessor	項番19を参照
38		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。	updateDAO	データベースBean定義ファイルを参照
39		chunkWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	blogic	ジョブBean定義ファイルを参照
40		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理実行クラスの定義。	blogicResultHandler	デフォルトBean定義ファイルを参照
41		blogicExecutor	jp.terasoluna.fw.batch.core.StandardBlogicExecutor	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
42		-	-	-	defaultExceptionHandler	デフォルトBean定義ファイルを参照
43		-	-	-	mbeanRegister	フレームワークBean定義ファイルを参照
44		-	-	-	transactionManager	データベースBean定義ファイルを参照
45		-	-	-	useSavepoint	デフォルトBean定義ファイルを参照
46	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。	partitionNo	初期値『-1』
47		JobStatus	jp.terasoluna.fw.batch.core.JobStatus	ジョブの処理状況用Bean定義。	defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
48		-	-	-	partitionNo	初期値『-1』
49		-	-	-	transactionManager	データベースBean定義ファイルを参照
50		-	-	-	useSavepoint	デフォルトBean定義ファイルを参照
51		-	-	-	partitionNo	初期値『-1』
52		-	-	-	defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
53		-	-	-	multiplicity	ジョブBean定義ファイルを参照
54		-	-	-	queryDAO	データベースBean定義ファイルを参照
55	ジョブリスタート管理テーブルハンドラ	threadSize	java.lang.Integer	スレッドプールのサイズ	updateDAO	データベースBean定義ファイルを参照
56		JobRestartTableHandler	jp.terasoluna.fw.batch.restart.JobRestartTableHandler	ジョブリスタート管理テーブルのハンドラクラス用Bean定義。	jobRestartInfoFactory	項番22を参照
57		-	-	-	なし	なし
58	ジョブリスタート情報用パラメータクラス	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。	なし	なし

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	親ジョブマネージャ	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
					workQueueFactory	項番2を参照
					preProcessor	項番3を参照
2	親ジョブマネージャ	-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応親ジョブ用作業キュー作成クラスの定義。	postProcessor	項番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 非同同期型起動用スレッドプール定義ファイルを参照
					queueProcessor	項番5を参照
					queueLength	デフォルトBean定義ファイルを参照
3	parentJobPreProcessor	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	
4	parentJobPostProcessor	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	
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。	name	初期値 『partitionNoTransactionQueueProcessor』
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。	worker	項番6を参照
					name	初期値『childJobManager』
					collector	ジョブBean定義ファイルを参照
					workQueueFactory	項番7を参照
7	子ジョブマネージャ	-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応子ジョブ用作業キュー作成クラスの定義。	preProcessor	項番8を参照
					postProcessor	項番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 非同同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
8	jobPreProcessor	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	
9	jobPostProcessor	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	
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。	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	項番11を参照
					biologicExecutor	項番12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
12	ワーカ	biologicExecutor	jp.terasoluna.fw.batch.core.StandardBiologicExecutor	ビジネスロジック実行クラスの定義。	batchUpdateProcessor	項番13を参照
					batchUpdateResultHandler	フレームワークBean定義ファイルを参照
					biologic	ジョブBean定義ファイルを参照
					biologicResultHandler	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。	exceptionHandlerMap	デフォルトBean定義ファイルを参照	
14	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。	defaultJobExceptionHandler	デフォルトBean定義ファイルを参照
					updateDAO	データアクセスBeanファイルを参照
					mbeanRegister	フレームワークBean定義ファイルを参照
					partitionNo	初期値『-1』
15	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。	defaultJobExitCodeMap	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』
					defaultJobExitCodeMap	デフォルトBean定義ファイルを参照
16	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	multiplicity	ジョブ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	親ジョブマネージャ	jobManager	baseManager (jp.terasoluna.fw.batch.core.JobManager)	親ジョブ実行用マネージャの定義。	name collector workQueueFactory preProcessor postProcessor	初期値 [partitionSingleTransactionJobManager] ジョブBean定義ファイルを参照 項番2参照 項番3を参照 項番5を参照
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応親ジョブ用作業キュー作成クラスの定義。	workerExecutorService queueProcessor queueLength multiplicity	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 項番7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
3		parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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	項番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.springsupport.transaction.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	項番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	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 worker	初期値 [partitionSingleTransactionQueueProcessor] 項番8を参照
8	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 collector workQueueFactory preProcessor postProcessor	初期値 [childJobManager] ジョブBean定義ファイルを参照 項番9参照 項番10を参照 項番12を参照
9		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応子ジョブ用作業キュー作成クラスの定義。	workerExecutorService queueProcessor queueLength	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 項番14を参照 デフォルトBean定義ファイルを参照
10		jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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	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
12		jobPostProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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	項番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	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 worker preProcessor postProcessor	初期値 [singleTransactionChildQueueProcessor] 項番17を参照 項番15を参照 項番16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	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	ワーカー	singleTransactionWorker	jp.terasoluna.fw.batch.core.JobWorker	チャンク処理クラスの定義。	blologicExecutor batchUpdateProcessor exceptionHandlerMap defaultJobExceptionHandler batchUpdateResultHandler	項番19を参照 項番18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
18		batchUpdateProcessor	jp.terasoluna.fw.batch.standard.StandardBatchUpdateProcessor	バッチ更新処理クラスの定義。	updateDAO	データアクセスBean定義ファイルを参照
19		blologicExecutor	jp.terasoluna.fw.batch.core.StandardBlologicExecutor	ビジネスロジック実行クラスの定義。	blologic blologicResultHandler exceptionHandlerMap defaultJobExceptionHandler	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
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。	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
21		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
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	multiplicity	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

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
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 『sequentialSingleTransactionJobManager』
					collector	ジョブBean定義ファイルを参照
					workQueueFactory	項番2を参照
					preProcessor	項番3を参照
					postProcessor	項番5を参照
2		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応親ジョブ用作業キュー作成クラスの定義。	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
					queueProcessor	項番7を参照
					queueLength	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
3	親ジョブマネージャ	parentJobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.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	項番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.springsupport.transaction.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	項番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	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 『sequentialSingleTransactionQueueProcessor』
					worker	項番8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
					workQueueFactory	項番9を参照
					preProcessor	項番10を参照
					postProcessor	項番12を参照
9		-	jp.terasoluna.fw.batch.commonj.WorkManagerTaskWorkQueueFactory	commonj対応子ジョブ用作業キュー作成クラスの定義。	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
					queueProcessor	項番14を参照
					queueLength	デフォルトBean定義ファイルを参照
					supportProcessor	項番11を参照
10	子ジョブマネージャ	jobPreProcessor	transactionalSupportProcessor (jp.terasoluna.fw.batch.springsupport.transaction.TransactionalSupportProcessor)	トランザクション管理有りの子ジョブ前処理の定義。		
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.springsupport.transaction.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	項番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		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 『singleTransactionChildQueueProcessor』
					worker	項番17を参照
					preProcessor	項番15を参照
					postProcessor	項番16を参照
15	子ジョブキュープロセッサ	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
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。	blologicExecutor batchUpdateProcessor exceptionHandlerMap defaultJobExceptionHandler batchUpdateResultHandler	項番19を参照 項番18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
18	フーカ	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
19		blologicExecutor	jp.terasoluna.fw.batch.core.StandardBLogicExecutor	ビジネスロジック実行クラスの定義。	blologic blologicResultHandler exceptionHandlerMap defaultJobExceptionHandler	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
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。	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
		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
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	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.TextGetString	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.common.DefaultBeanValidator	バリデータのサンプルBean定義。	validatorFactory	項番11を参照
11	ファイル入力チェック（バリデータファクトリの記述例）	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）	-	-