

11 問題判別	2
11-1 IBM SUPPORT ASSISTANT WORKBENCH 4.1 の概要	2
11-2 IBM SUPPORT ASSISTANT 5.0 TEAM SERVER の概要	4
11-3 IBM SUPPORT ASSISTANT DATA COLLECTOR	5
11-4 集中ロギング	6
11-5 クロス・コンポーネント・トレース(XCT)	7

11 問題判別

この章では、IBM が提供する問題判別ツールである IBM Support Assistant(以下、ISA)と、WAS8.5 で使用できる問題判別用の機能について説明します。

ISA には問題判別用のクライアント・サイド・アプリケーションとして ISA Workbench 4.1 とサーバー・ベースのアプリケーションとして ISA 5.0 Team Server、情報収集用のツールとして ISA Data Collector があります。、WAS8.5 では ISA Workbench 4.1.2、ISA 5.0 Team Server がサポートされています。

WAS8.5 の新機能として、4-4 で集中ロギング、4-5 でクロス・コンポーネント・トレースについて説明します。

11-1 IBM Support Assistant Workbench 4.1

の概要

ISA Workbench は、WAS 上で発生した問題についてユーザーが診断するためのツールです。さまざまなアドオンが用意されており、ファイルの収集を自動化したり、成果物を分析するツールを使用したりすることで、エラーの根本原因を把握することができます。

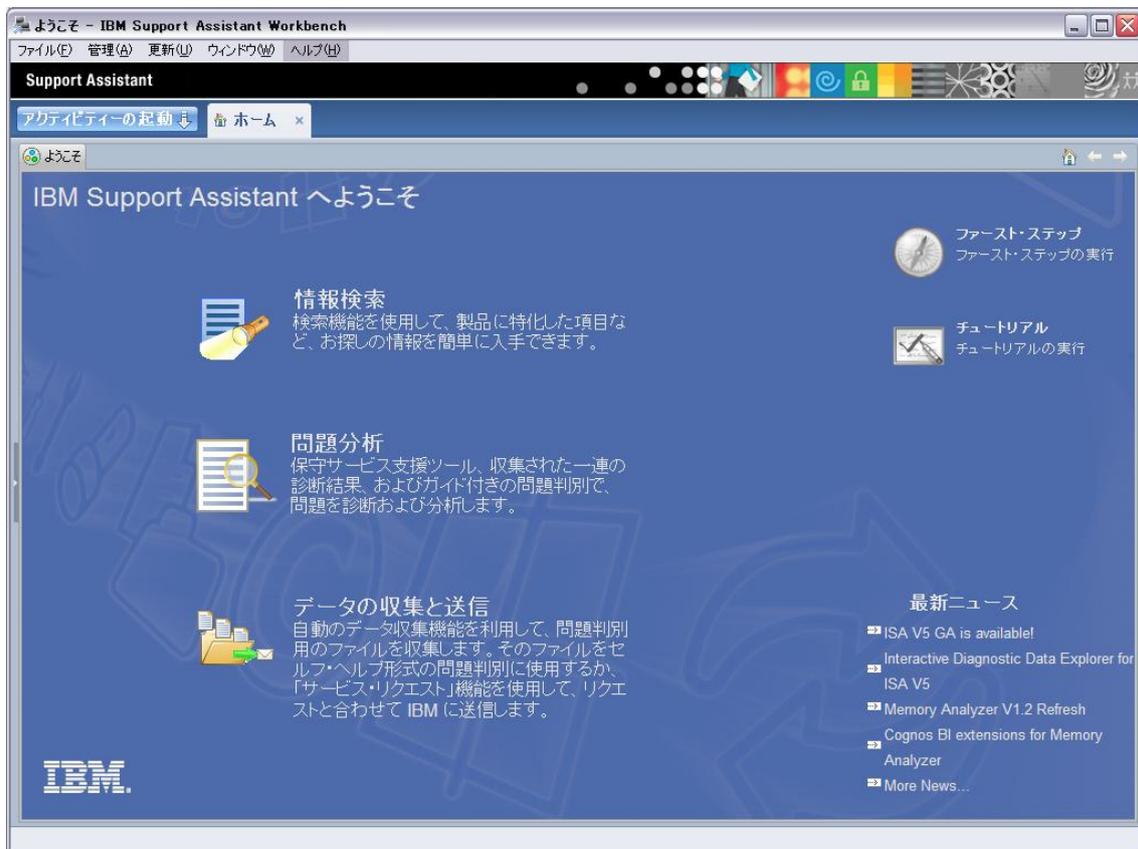


図 11-1 ISA WorkBench

以下からダウンロード頂けます。

IBM Support Assistant Workbench

<http://www.ibm.com/software/support/isa/workbench.html>

ISA Workbench の導入方法、および使用方法については以下のガイドを参照ください。

IBM Support Assistant 4.1 利用ガイド

http://www.ibm.com/developerworks/jp/websphere/library/was/isa41_guide/

ISA Workbench で使用可能なアドオンについては以下を参照ください。

IBM Support Assistant Workbench 4.1 Tool Add-Ons

<http://www.ibm.com/support/docview.wss?uid=swg27013116>

ここでは、ISA Workbench で使用できるアドオンについていくつかご紹介致します。

IBM Thread and Monitor Dump Analyzer (以下、TMDA)

TMDA は javacore を解析し、デッドロックやスレッドハングなどの根本原因を調査するツールになります。各 javacore を比較して、ガーベッジ・コレクションの頻度や、割り振りの失敗頻度、およびハング疑いのリストを表示したりすることができます。前述の ISA 4.1 利用ガイドの中で操作方法について説明されています。

IBM Monitoring and Diagnostic Tools for Java - Health Center

Java の稼働状況のモニタリングやプロファイリングに使用するツールになります。こちらについては利用ガイドも発行されておりますのでご参照ください。

Health Center 利用ガイド

http://www.ibm.com/developerworks/jp/websphere/library/was/hc_guide/

IBM Pattern Modeling and Analysis Tool for Java Garbage Collector (以下、PMAT)

PMAT は verbose GC ログの解析ツールになります。Java ヒープの微調整や、メモリーリークが起きていないか調査するために使用することができます。

IBM Monitoring and Diagnostic Tools for Java - Garbage Collection and Memory Visualizer (以下、GCMV)

GCMV も verbose GC ログの解析ツールになります。データをグラフや表形式で表示することで問題判別や Java プロセスの調整に使用することができます。

11-2 IBM Support Assistant 5.0 Team Server

の概要

ISA Team Server は、ISA Workbench と同様に問題判別用のツールとなりますが、WAS が稼働する各サーバーに Agent を導入し AgentManager の管理の下で接続を行っていた ISA Workbench とは異なり、JVM 上でアプリケーションとして稼働します。このためユーザーはブラウザからログをアップデートしたり、各種ツールにアクセスすることができます。

ISA Team Server で使用可能な問題判別ツールについては以下を参照下さい。

Problem Determination Tools for IBM Support Assistant 5

<http://www.ibm.com/support/docview.wss?uid=swg27036217>

ISA 5.0 Team Server には以下の 3 つの導入方法があります。

- ▶ IBM Installation Manager を使用した導入 (推奨)
- ▶ EAR としてインストール
- ▶ 圧縮ファイルによるインストール

IBM Installation Manager を使用した導入 (推奨)

ISA 5.0 Team Server を IBM Installation Manager (以下、IIM) を使用して導入することができます。IIM を使用した導入では、ISA のバージョンアップが可能のため推奨される導入手順となっています。IIM1.5.2 以降の使用がサポートされます。

以下からダウンロード頂けます。

<http://www.ibm.com/support/docview.wss?uid=swg27039277>

EAR としてインストール

既に稼働している WAS 上で ISA Team Server をデプロイすることが可能です。システム要件として WAS7.0.0.21 以降をサポートしています。

この方法では個別に EAR という形では提供されていないため、IIM から EAR デプロイメントを選択することで ISA5.ear を取得します。

以下からダウンロード頂けます。

<http://www.ibm.com/support/docview.wss?uid=swg27039275>

圧縮ファイルによるインストール

Liberty ベースの組み込み WAS と ISA Team Server を圧縮ファイルという形式でダウンロードできます。この導入方法ではバージョンアップが出来ないため、今後アップデートがあった場合は再度導入しなおす必要があります。

以下からダウンロード頂けます。

<http://www.ibm.com/support/docview.wss?uid=swg27039274>

11-3 IBM Support Assistant Data Collector

WAS7.0 で非推奨となった collector コマンドに代わり、WAS8.5 では IBM Support Assistant Data Collector (以降、ISA Data Collector) を使用することができます。ISA Data Collector は問題データの自動収集を行うツールです。様々な問題に対する症状の分析の支援を行い、IBM サポートへ送付するためにも使用できます。

このツールは、WAS に付属する isadc コマンドを実行する他、ブラウザからの実行も可能です。

WAS に付属する isadc コマンドを実行

<WAS_Install_Root>/bin 配下、または、<Profile_Root>/bin 配下の isadc コマンドを実行することで、ツールがコンソール・モードで実行されます。

詳しくは、Information Center の情報を参照してください。

WAS Information Center 「IBM Support Assistant Data Collector の使用」:

http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.nd.multiplatform.doc/ae/trb_isadestart.html

ブラウザから実行する

以下の ISA Data Collector のページにアクセスし、"Select a brand to begin the download process"で "WebSphere"、"Select a product to access the download package"で "WebSphere Application Server" を選択します。"NEW: Try the new Web-enabled collector" のリンクをクリックすることで Data Collector アプレットが起動します。

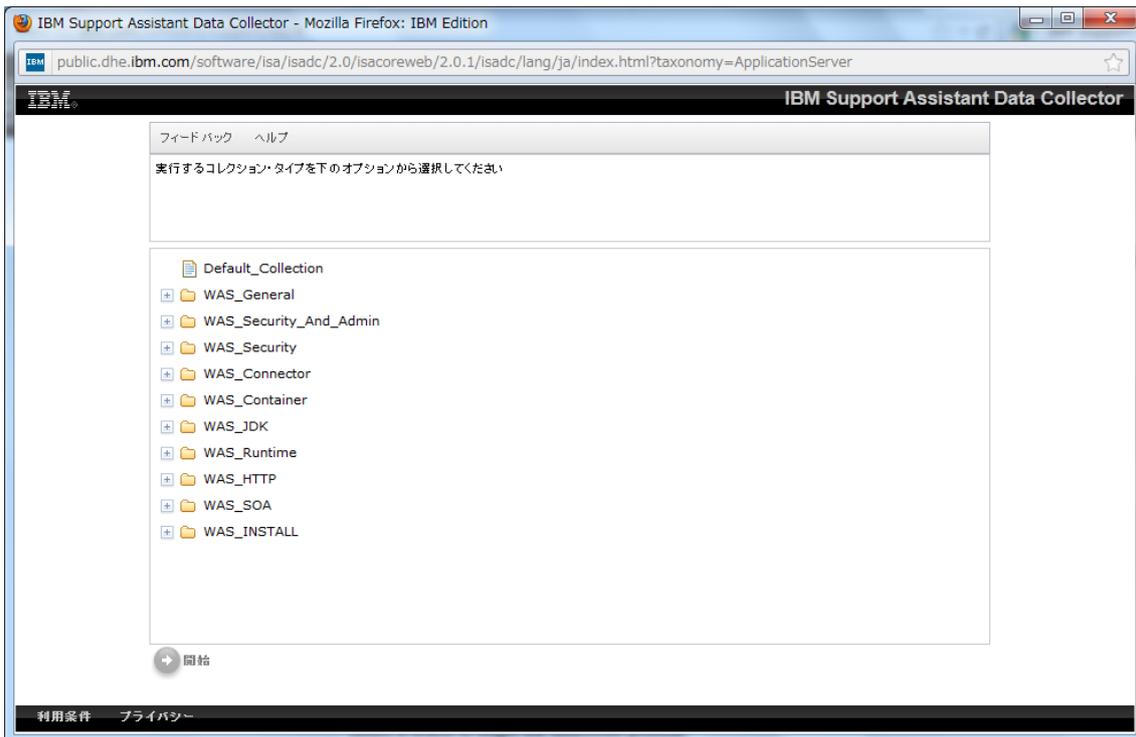


図 11-2 ISA Data Collector

IBM Support Assistant Data Collectors

<http://www.ibm.com/software/support/isa/>

11-4 集中ロギング

集中ロギングは、問題判別の際に必要な情報を収集するためのツールです。パフォーマンス上の理由により常時使用可能にするのではなく、特定の問題をキャプチャーするために用いるツールとなります。集中ロギングには `mustGather` 文書のタイプを指定する方法と、要求ベースのトレース・ルールを指定する方法の 2 つがあります。

mustGather 文書タイプを指定

`mustgather.py` は、`enable/collect/disable` のコマンドと、引数に `mustGather` 文書タイプを指定して利用します。タイプは以下の種類をサポートしています。

- ▶ 404 : 404 HTTP 応答コード
- ▶ 503 : 503 HTTP 応答コード
- ▶ 504 : 504 HTTP 応答コード
- ▶ agent : ノード・エージェント
- ▶ appedition : アプリケーション・エディション・マネージャー
- ▶ arfm : オートノミック要求フロー・マネージャー
- ▶ dc : 動的クラスター

- ▶ hadmgr : 高可用性デプロイメント・マネージャー
- ▶ hmm : 正常性モニター
- ▶ odr : オンデマンド・ルーター
- ▶ operations : 「Extended Deloyment」および「オペレーション」タブでの可視化の問題
- ▶ reports : 「レポート」タブでの可視化の問題
- ▶ reportsPerf : 「レポート」タブに表示されるパフォーマンス・データの可視化の問題
- ▶ repository : 拡張リポジトリ・サービス
- ▶ sip : SIP 要求ルーティング

例:404HTTP ステータスコードに関する情報を収集する場合

- ① 有効化: `wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py enable 404`
- ② 問題を再現
- ③ 収集: `wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py collect 404 "c:\mydocs\collection.zip"`
- ④ 無効化: `wsadmin -lang jython -f c:\WebSphere\AppServer\bin\mustgather.py disable 404`

要求ベースのトレース・ルールを指定

要求ベースのトレース・ルールは 1 つの式とルール ID と呼ばれる ID で構成されています。オプションで、ルールに ODR トレース・ストリングおよびアプリケーション・サーバー・トレース・ストリングを指定することもできます。ログ・ファイルには 1 つ以上のルールに一致する要求の開始マーカーと終了マーカーが配置されます。

- ① 有効化: `wsadmin.sh -lang jython -f setupReqBasedTracing.py enableReqBasedTracing -ruleExpression:<expression> -odrTraceSpec:<trace strings> -appServerTraceSpec:<trace string> -ruleID:<rule ID>`
- ② 無効化: `wsadmin.sh -lang jython -f setupReqBasedTracing.py disableReqBasedTracing -ruleIDs:<rule ID1>,<rule ID2>...,<rule IDn>`

詳しくは、Information Center の情報を参照してください。

WAS Information Center 「HTTP オペランド」:

http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.wve.doc/ae/rwve_odrhttp.html

11-5 クロス・コンポーネント・トレース(XCT)

クロス・コンポーネント・トレースは、ログ・エン트리とトレース・エントリに注釈をつけることで両者の関連付けを可能にする機能です。また、サーバーが High Performance Extensible Logging (HPEL) を使用するよう構成されている場合には、ログに XCT 要求 ID (requestID) を追加し、1 つのリクエストに

関連するログを識別するために使用できます。スレッドやプロセスをまたぐとき、または WAS 間で転送される際に、そのフローをたどることが可能になり、リクエストの失敗に関連するコンポーネントを洗い出すのに役立ちます。

クロス・コンポーネント・トレースは、管理コンソールもしくは `wsadmin` コマンドによって構成します。`wsadmin` コマンドを使用する方法には、AdminConfig オブジェクトを使用する方法と AdminControl オブジェクトを使用する方法の 2 つがあります。AdminConfig オブジェクトを使用した設定方法では、サーバー再起動後に設定が有効となります。

管理コンソールを使用した構成

- ステップ1: 管理コンソールを起動し、[トラブルシューティング]→[ログおよびトレース]とクリックします。次に、XCT を使用可能にするサーバーを選択し、[ログ詳細レベルの変更]とクリックします。
- ステップ2: [ログとトレースの相関を有効にする]に、チェックを入れます。

図 11-3 XCT 設定画面

- ステップ3: 必要に応じて、[要求 ID をログ・レコードとトレース・レコードに組み込む]、[要求 ID をログ・レコードとトレース・レコードに組み込み、関連ログ・レコードを作成する]、または[要求 ID をログ・レコードとトレース・レコードに組み込み、関連ログ・レコードを作成し、データ・スナップショットを取り込む]を選択します。
- ステップ4:[OK]ボタンをクリックし、[保存]します。
- ステップ5:アプリケーション・サーバーを再始動します。

AdminConfig オブジェクトを使用した構成

- ステップ1:サーバーの XCT を使用可能にする
サーバーが基本ログおよびトレース・モードを使用している場合は、HighPerformanceExtensibleLogging の代わりに RASLoggingService を指定します。

```
# get rid of existing property if already present
configId = AdminConfig.getid("/Cell:myCell/Node:myNode/Server:myServer/HighPerformanceExtensibleLogging:/Property:com.ibm.websphere.logging.enableCorrelation") if
(len(configId) > 0):
AdminConfig.remove(configId)
# add new property
LoggingService = AdminConfig.getid("/Cell:myCell/Node:myNode/Server:myServer/HighPerformanceExtensibleLogging:/")
AdminConfig.create("Property", LoggingService, [{"name", "com.ibm.websphere.logging.enableCorrelation"}, {"value", "true"}])
AdminConfig.save()
```

- ステップ2:XCT のログの設定を変更する
サーバーが基本ログおよびトレース・モードを使用している場合は、HighPerformanceExtensibleLogging の代わりに RASLoggingService を指定します。

```
# get rid of existing property if already present
configId = AdminConfig.getid("/Cell:myCell/Node:myNode/Server:myServer/HighPerformanceExtensibleLogging:/Property:com.ibm.websphere.logging.correlationLevel") if
(len(configId) > 0):
AdminConfig.remove(configId)
# add new property
LoggingService = AdminConfig.getid("/Cell:myCell/Node:myNodeServer:myServer/HighPerformanceExtensibleLogging:/")
AdminConfig.create("Property", LoggingService, [{"name", "com.ibm.websphere.logging.correlationLevel"}, {"value", "LOG"}])
AdminConfig.save()
```

- ステップ3:設定を反映するためにサーバーの再起動を行います。

AdminControl オブジェクトを使用した構成

- ステップ1:サーバーの XCT を使用可能にする
サーバーが基本のログおよびトレース・モードを使用している場合は、サーバーが稼働していることを確認して、HPELControlService の代わりに RasLoggingService を指定します。

```
LoggingMBean = AdminControl.queryNames(' cell=myCell, node=myNode, type=HPELControlService, process=myServer, *')  
AdminControl.setAttribute(LoggingMBean, "correlationEnabled", "true")
```

- ステップ2:XCT のログの設定を変更する
サーバーが基本のログおよびトレース・モードを使用している場合は、サーバーが稼働していることを確認して、HPELControlService の代わりに RasLoggingService を指定します。

```
LoggingMBean = AdminControl.queryNames(' cell=myCell, node=myNode, type=HPELControlService, process=myServer, *')  
AdminControl.setAttribute(LoggingMBean, "xctLevel", "LOG")
```

HPEL ログ・リポジトリーの内容の照会には LogViewer コマンドを使用しますが、この XCT 要求 ID でフィルターを掛けることが可能です。以下はそのサンプルになります。

```
logViewer.sh -includeExtensions requestID=a856cb2c-79ed-4d62-a3cf-a9908b2db07b
```

詳しくは、Information Center の情報を参照してください。

WAS Information Center 「LogViewer コマンド行ツール」:

http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.nd.multiplatform.doc/ae/rtrb_logviewer.html