最終更新日 ： 2010/06/25

TOPPERS/TLVマニュアル

（1.2 対応）

名古屋大学 大学院情報科学研究科
附属組込みシステム研究センター

はじめに

・本マニュアルは，TLVの基本的な使い方及び，TOPPERS/ASPカーネルまたは，TOPPERS/FMPカーネルでTLVで可視化するためのログ取得方法について解説している.

・ログの標準形式への変換ルールに関しては，TLV_convert_rules.ppt を参照のこと.

・ログの可視化のための可視化変換ルールについては，TLV_visualize_rules.ppt を参照のこと.

目次

- ファイル一覧
- 実行環境
- TOPPERS/ASPカーネルとTOPPERS/FMPカーネルのトレースログの取得
- 機能紹介

ファイル一覧

TOPPERS/TLVパッケージのファイル構成

フォルダ名	ファイル名	説明
/	README.txt	TLVの簡単な紹介
	TraceLogVisualizer.exe	TLVの本体
doc/	TLV.pdf	TLV本体のマニュアル
	TLV_convert_rules.pdf	変換ルールのマニュアル
	TLV_visualize_rules.pdf	可視化ルールのマニュアル
convertRules/	asp.cnv	TOPPERS/ASPカーネルの変換ルール
	fmp.cnv	TOPPERS/FMPカーネルの変換ルール
	tecs.cnv	TECSの変換ルール
resourceHeaders/	asp.resh	TOPPERS/ASPカーネルのリソースファイル
	fmp.resh	TOPPERS/FMPカーネルのリソースファイル
	tecs.resh	TECSのリソースファイル
visualizeRules/	asp_rules.viz	TOPPERS/ASPカーネルの可視化ルール
	asp_shapes.viz	TOPPERS/ASPカーネルの図形定義
	fmp_rules.viz	TOPPERS/FMPカーネルの可視化ルール
	fmp_shapes.viz	TOPPERS/FMPカーネルの図形定義
	tecs.viz	TECSの可視化ルール
	toppers_rules.viz	TOPPERS共通の可視化ルール
	toppers_shapes.viz	TOPPERS共通の図形定義

フォルダ名		ファイル名	説明
sampleFiles/	asp/	asp_short.log asp_short.res asp_short.tlv asp_long.log asp_long.res asp_long.tlv	TOPPERS/ASPカーネルのトレースログのサンプル
	fmp/	fmp_short.log fmp_short.res fmp_short.tlv fmp_long.log fmp_long.res fmp_long.tlv	TOPPERS/FMPカーネルのトレースログのサンプル
	tecs/	tecs.log tecs.res tecs.tlv	TECSのトレースログのサンプル
logtrace/	asp/	kernel_fncode.h tlv.tf trace_config.c trace_config.h trace_dump.c	TOPPERS/ASPカーネルのログトレースモジュール
	fmp/	tlv.tf	TOPPERS/FMPカーネルのログトレースモジュール

実行環境

実行環境

TLV実行環境

- WindowsXP / Vista
- Microsoft .NET Framework 3.5　をインストールすること.

ログ取得対象

- TOPPERS/ASPカーネル 1.3.2/1.3.1のトレースログ
- TOPPRES/FMPカーネル 0.B.0 のトレースログ
- TECSのログ

TOPPERS/ASPカーネルと TOPPERS/FMPカーネルの トレースログの取得

TOPPERS/ASPカーネルのトレースログの取得

・TLVパッケージのlogtrace/asp 以下のファイルをカーネルの asp/arch/logtrace に置く.

・asp/kernel/kernel.tf の最後に以下を追加.

```
最後の行：$INCLUDE"arch/logtrace/tlv.tf"$
```

・対象プログラムのMakefileを編集してトレースログを有効にする.

```
92行目　：ENABLE_TRACE = true
```

TOPPERS/ASPカーネルのトレースログの取得

- asp/doc/user.txt の 10.3トレースログ記録のサンプルコードの使用方法を参照して，ログの取得と出力を行う．
- トレースログ記録の使用方法の一例として，システム起動時にトレースログの記録を開始し，システム終了時に記録したトレースログをダンプするためには，システムコンフィギュレーションファイル(.cfg)に次のような記述を追加する．

```
#include "logtrace/trace_config.h"
ATT_INI({ TA_NULL, TRACE_AUTOSTOP, trace_initialize });
ATT_TER({ TA_NULL, target_fput_log, trace_dump });
```

- ここで，初期化ルーチン（trace_initialize）への引数は，初期化直後のトレースログの動作モードを指定するものである．指定できる動作モードについては，arch/logtrace/trace_config.h中のコメントに説明がある．
- 終了処理ルーチン（trace_dump）は，記録されたトレースログをターゲット依存の低レベル出力機能（target_fput_log）を利用してダンプするためのものである．トレースログを別の方法で取り出す場合には，終了処理ルーチンを登録する必要はない．

TOPPERS/ASPカーネルのトレースログの取得

trace_config.h をログを取得する環境に合わせて変更する

・バッファサイズ
  ・TCNT_TRACE_BUFFER

・時刻取得ルーチン
  ・ターゲット依存で時刻を取得したい場合は，TRACE_GET_TIM に定義する.

・取得するログトレース
  ・取得したいログトレース以外はコメントアウトする.
  ・例）loc_cpuのログを取らない場合は，LOG_LOC_CPU_ENTER()とLOG_LOC_CPU_LEAVE(ercd)のマクロをコメントアウトする.

```
//#define LOG_LOC_CPU_ENTER()
//#define LOG_LOC_CPU_LEAVE(ercd)
```

TOPPERS/ASPカーネルのトレースログの取得

各状態表示のための必須のログトレース

- タスクの状態表示
  - LOG_TSKSTAT(p_tcb)
- ハンドラの実行表示
  - LOG_INH_ENTER(inhno)/LOG_INH_LEAVE(inhno),
    LOG_ISR_ENTER(intno)/LOG_ISR_LEAVE(intno),
    LOG_CYC_ENTER(p_cyccb)/LOG_CYC_LEAVE(p_cyccb),
    LOG_ALM_ENTER(p_almcb)/LOG_ALM_LEAVE(p_almcb),
    LOG_EXC_ENTER(excno)/LOG_EXC_LEAVE(excno),
    LOG_TEX_ENTER(texptn)/LOG_TEX_LEAVE(texptn)
- システムコール表示
  - 有効にしたログトレースが表示される.

TOPPERS/FMPカーネルのトレースログの取得

・TLVパッケージのlogtrace/fmp 以下のファイルを
カーネルの fmp/arch/logtrace に置く.

・fmp/kernel/kernel.tf の最後に以下を追加.

最後の行：$INCLUDE"arch/logtrace/tlv.tf"$

・対象プログラムのMakefileを編集してトレースログ
を有効にする.

•92行目 ：ENABLE_TRACE = true

TOPPERS/FMPカーネルのトレースログの取得

- fmp/doc/user.txt の 10.3トレースログ記録のサンプルコードの使用方法を参照して，ログの取得と出力を行う．
- トレースログ記録の使用方法の一例として，システム起動時にトレースログの記録を開始し，システム終了時に記録したトレースログをダンプするためには，システムコンフィギュレーションファイル(.cfg)に次のような記述を追加する．

```
#include "logtrace/trace_config.h"
ATT_INI({ TA_NULL, TRACE_AUTOSTOP, trace_initialize });
ATT_TER({ TA_NULL, target_fput_log, trace_dump });
```

- ここで，初期化ルーチン（trace_initialize）への引数は，初期化直後のトレースログの動作モードを指定するものである．指定できる動作モードについては，arch/logtrace/trace_config.h中のコメントに説明がある．

- 終了処理ルーチン（trace_dump）は，記録されたトレースログをターゲット依存の低レベル出力機能（target_fput_log）を利用してダンプするためのものである．トレースログを別の方法で取り出す場合には，終了処理ルーチンを登録する必要はない．

TOPPERS/FMPカーネルのトレースログの取得

trace_config.h をログを取得する環境に合わせて変更する.

・バッファサイズ
  ・TCNT_TRACE_BUFFER

・時刻取得ルーチン
  ・ターゲット依存で時刻を取得したい場合は, TRACE_GET_TIM に定義する.

・取得するログトレース
  ・取得したいログトレース以外はコメントアウトする.
  ・例) loc_cpuのログを取らない場合は, LOG_LOC_CPU_ENTER()とLOG_LOC_CPU_LEAVE(ercd)のマクロをコメントアウトする.

```
//#define LOG_LOC_CPU_ENTER()
//#define LOG_LOC_CPU_LEAVE(ercd)
```

TOPPERS/FMPカーネルのトレースログの取得

各状態表示のための必須のログトレース

- タスクの状態表示
  - LOG_TSKSTAT(p_tcb)
- ハンドラの実行表示
  - LOG_INH_ENTER(inhno)/LOG_INH_LEAVE(inhno),
    LOG_ISR_ENTER(intno)/LOG_ISR_LEAVE(intno),
    LOG_CYC_ENTER(p_cyccb)/LOG_CYC_LEAVE(p_cyccb),
    LOG_ALM_ENTER(p_almcb)/LOG_ALM_LEAVE(p_almcb),
    LOG_EXC_ENTER(excno)/LOG_EXC_LEAVE(excno),
    LOG_TEX_ENTER(texptn)/LOG_TEX_LEAVE(texptn)
- システムコール表示
  - 有効にしたログトレースが表示される.

TLVへの入力ファイル

- TLVへの入力ファイルは，以下の二つであり，リソースファイルは，アプリケーションのビルド時に，トレースログファイルはアプリケーションの実行後に作成される.
  - リソースファイル（kernel.res）
    - アプリケーションをビルドすると，コンフィギュレーターにより，ビルドディレクトリに生成される.
  - トレースログファイル(xxx.log)
    - トレースログの出力をユーザーがファイルに保存する.

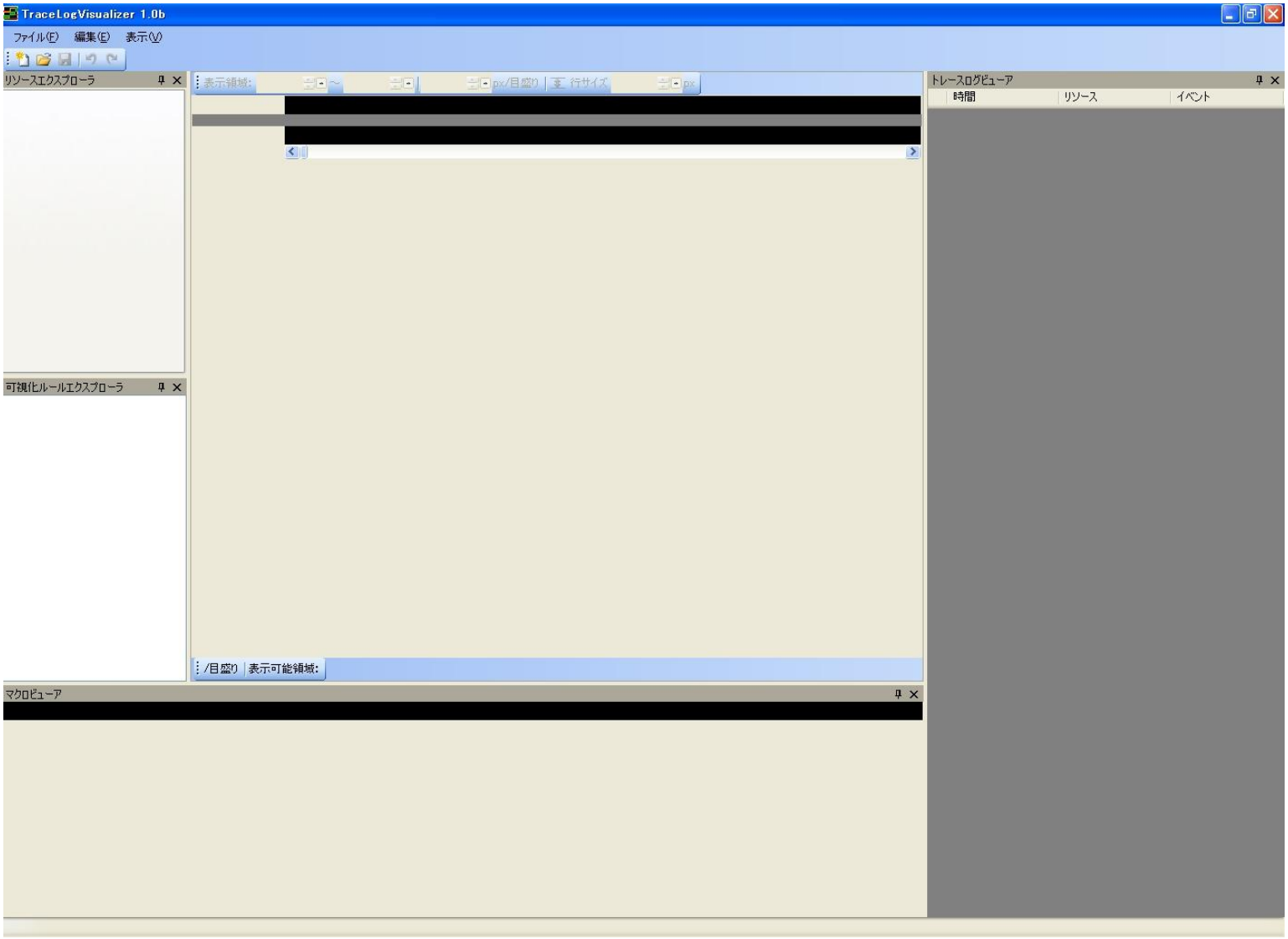
機能紹介

機能紹介

・sampleフォルダに入っているTOPPERS/ASPカーネルのリソースファイルとログファイルを用いて，TLVの機能を紹介する．

TLV実行

TLV初期画面



TLVメニュー一覧

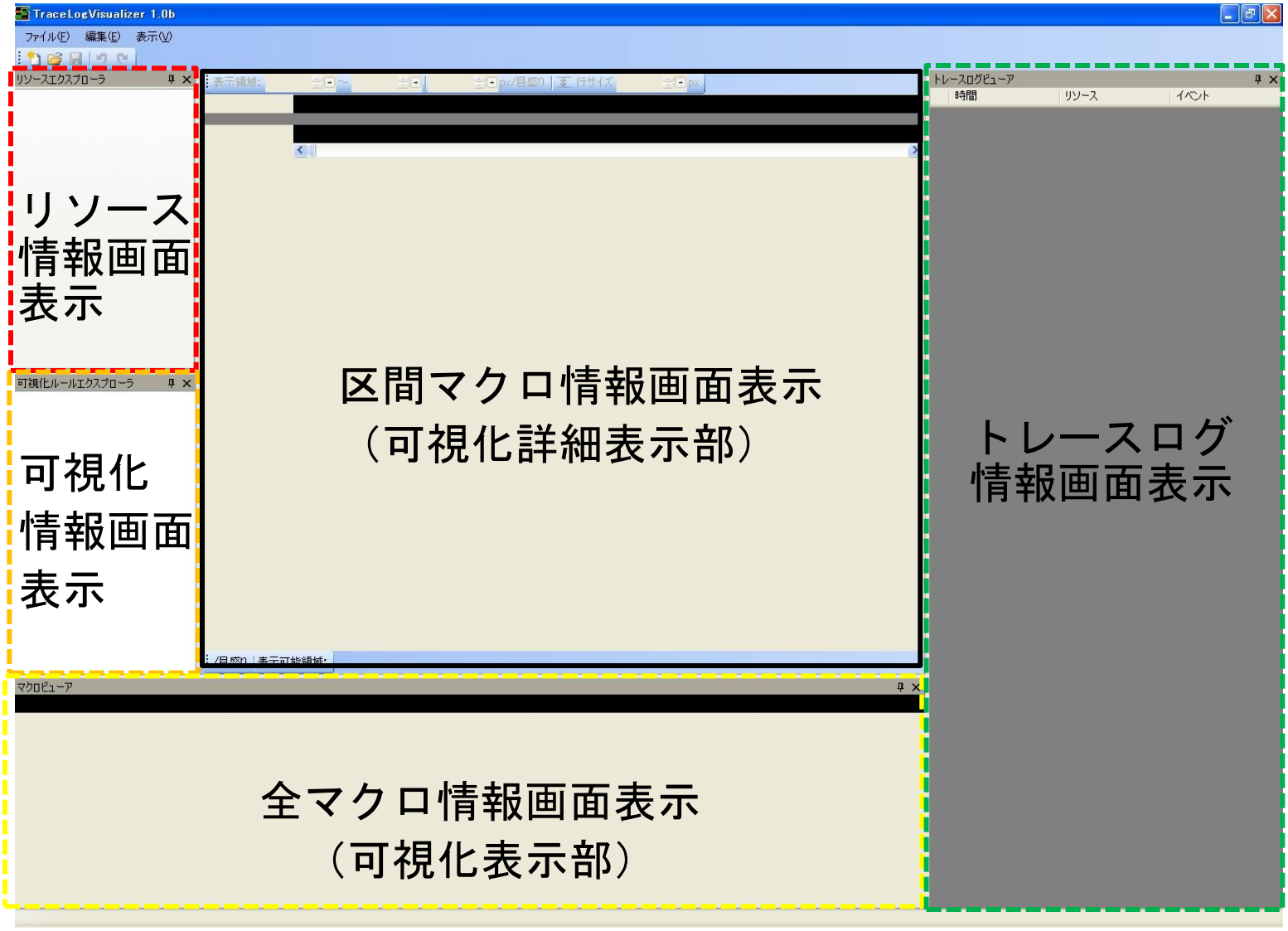
・ファイル
  ・新規作成－新しいログファイルをオープン
  ・開く－既存のログファイルをオープン(xxx.tlv)

・編集

・表示
  ・トレースログビュアー　－トレースログ画面表示
  ・リソースエクスプローラ　－リソース情報画面表示
  ・可視化ルールエクスプローラ－可視化情報画面表示
  ・マクロビュアー(可視化表示部)－全マクロ画面表示
  ・可視化詳細表示部　－　区間マクロ情報画面表示

TLV実行

TLV初期画面



ログファイルオープン

・TLV1.0rcログファイルオープンには、二つの方法がある.

１．新規作成

kernel.res、xxx.log　ファイルが必要.

「メニュー」→「ファイル」→「新規作成」をクリック

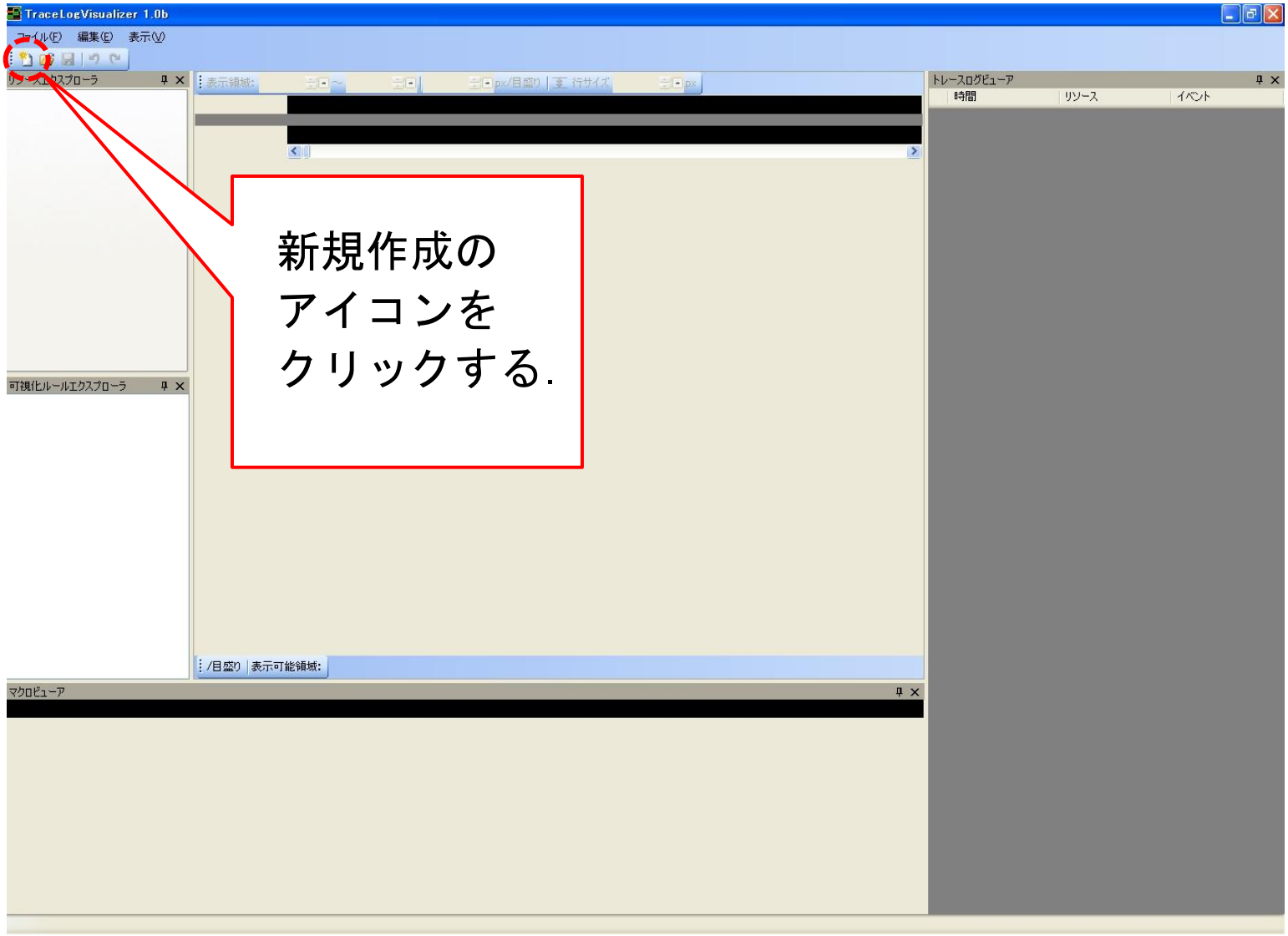
２．開く（保存したものを開く）

xxx.tlv　ファイルが必要.

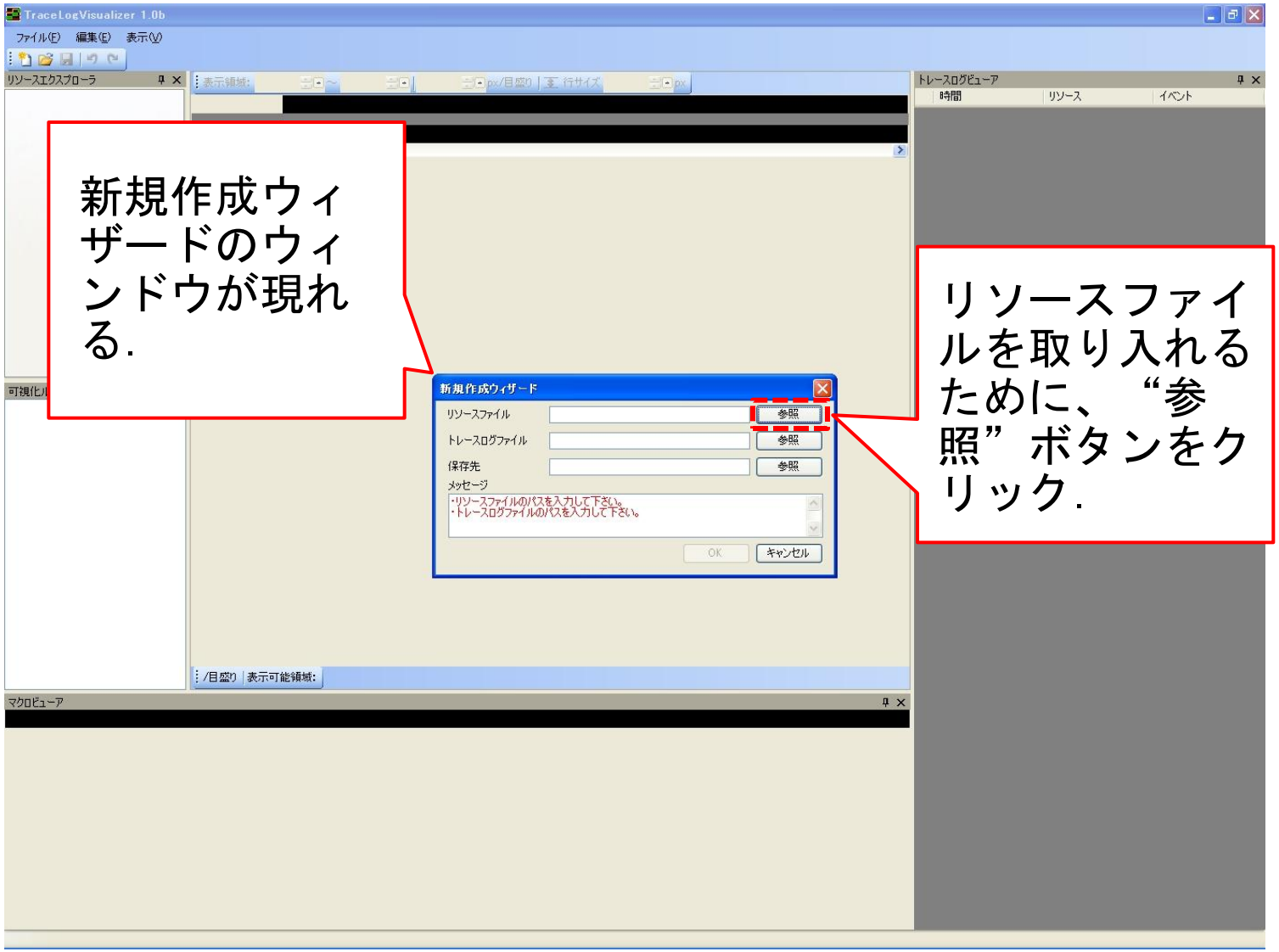
「メニュー」→「ファイル」→「開く」をクリック

1．新規作成ースタート

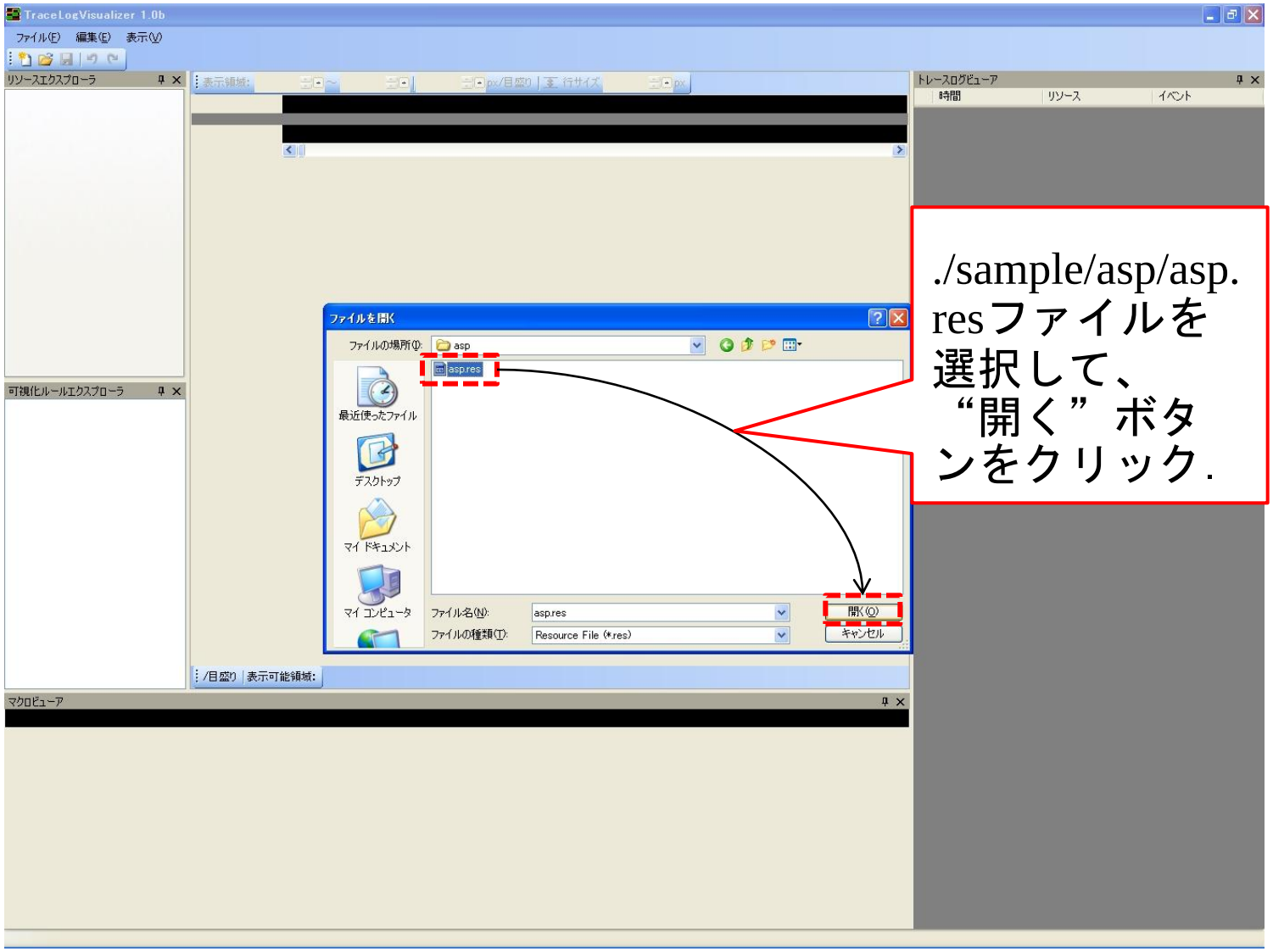
TLV初期画面



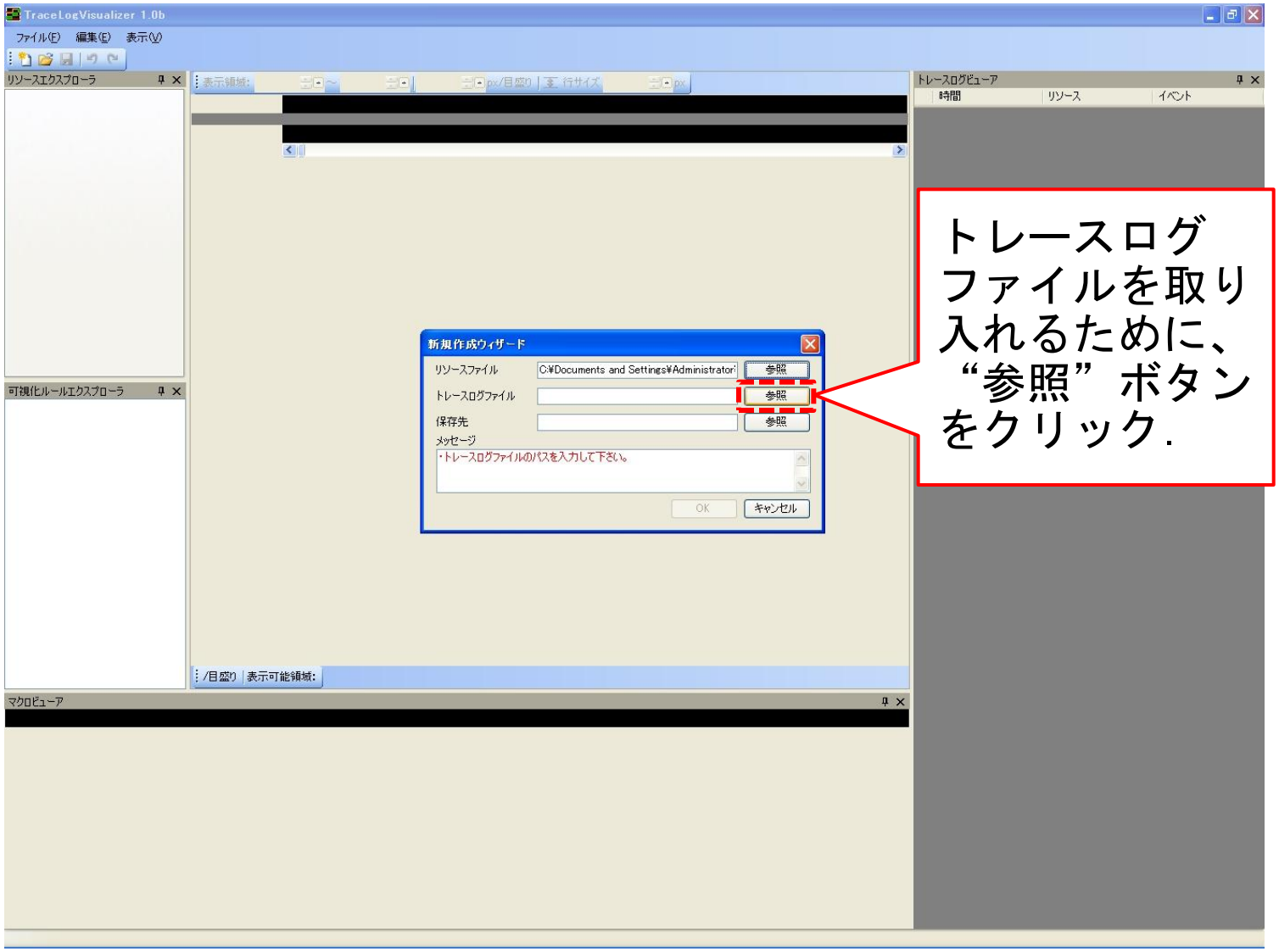
新規作成ーリソースファイルの参照



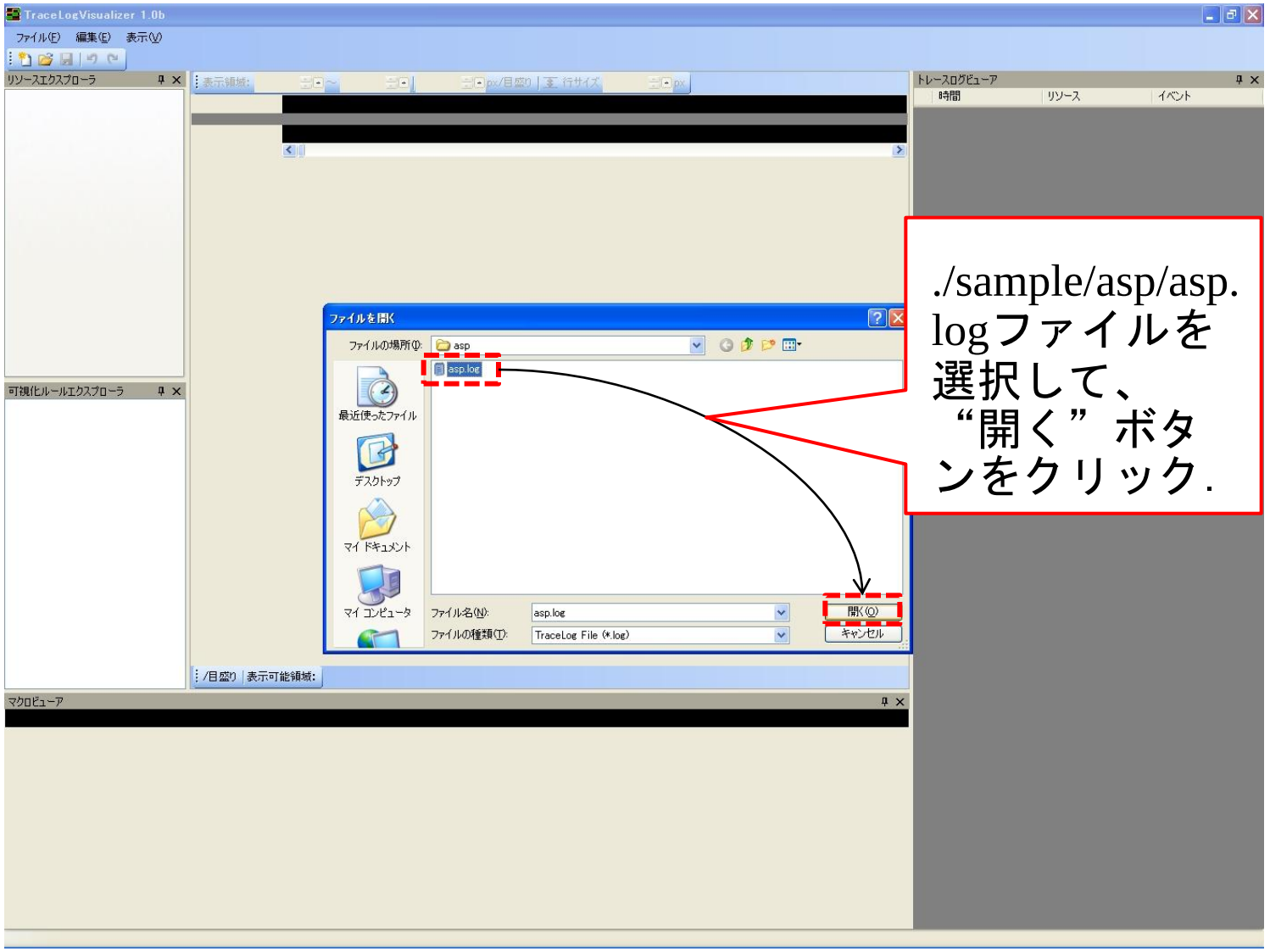
新規作成ーリソースファイルの参照



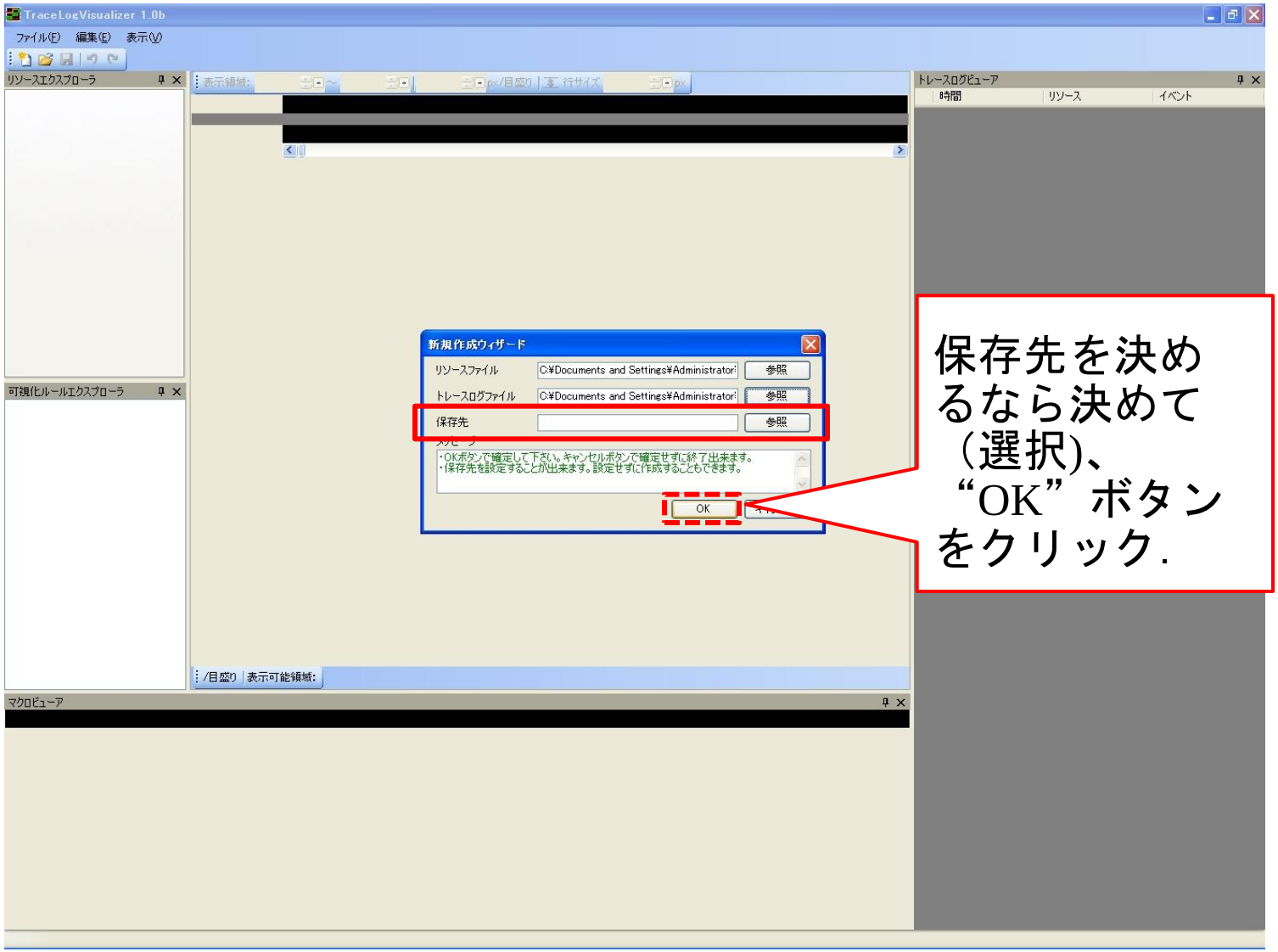
新規作成ー トレースログファイルの参照



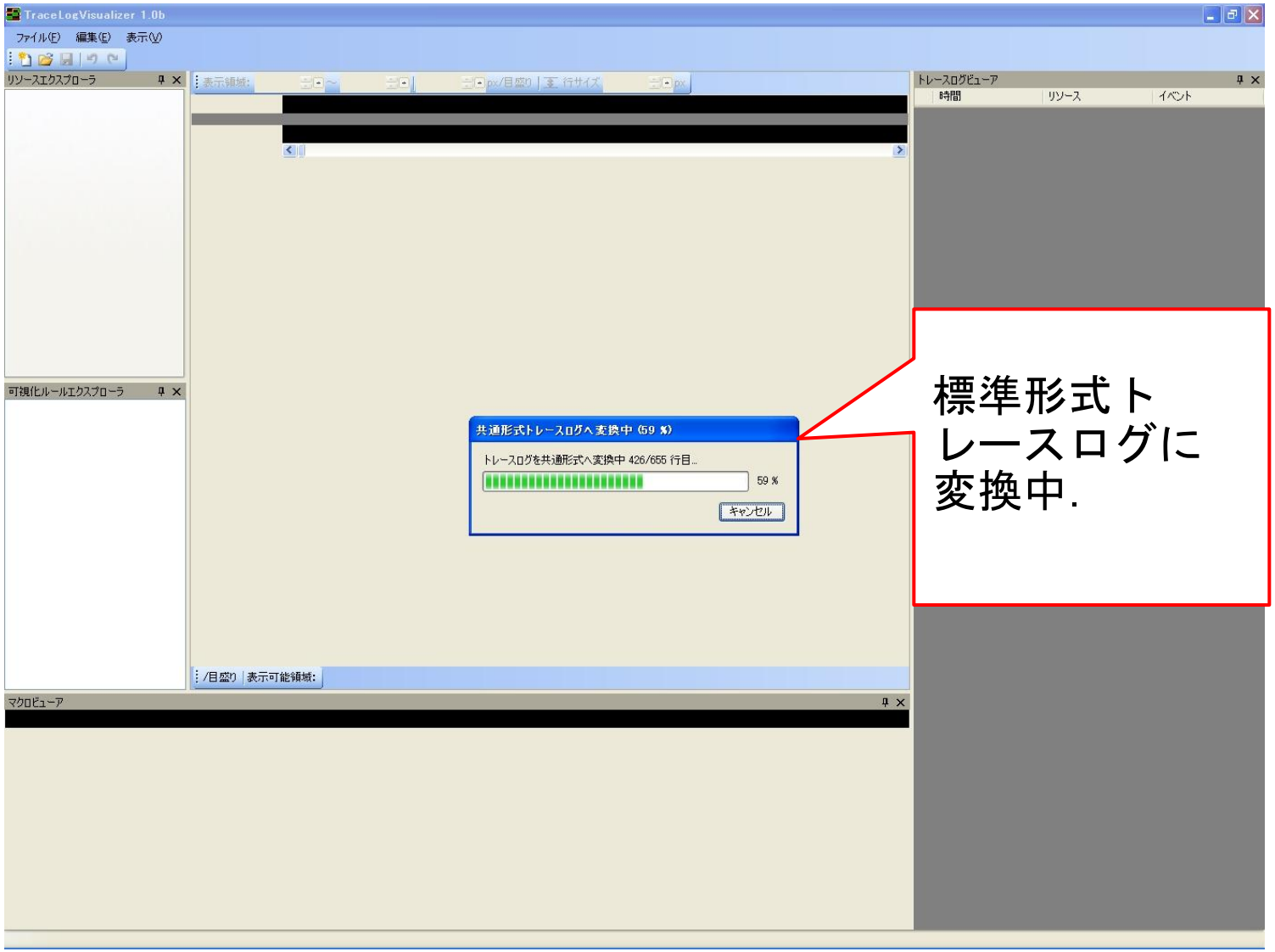
新規作成ー トレースログファイルの参照



新規作成－保存先の設定


TraceLogVisualizer 1.0b
ファイル(F)　編集(E)　表示(V)
リソースエクスプローラ
可視化ルールエクスプローラ
トレースログビューア
時間
リソース
イベント
マクロビューア
新規作成ウィザード
リソースファイル
C:¥Documents and Settings¥Administrator
参照
トレースログファイル
C:¥Documents and Settings¥Administrator
参照
保存先
参照
・OKボタンで確定して下さい。キャンセルボタンで確定せずに終了出来ます。
・保存先を設定することが出来ます。設定せずに作成することもできます。
OK
保存先を決めるなら決めて（選択)、“OK”ボタンをクリック.

新規作成－標準形式トレースログに変換



ログファイルオープン

・TLV1.0ログファイルオープンには二つの方法がある.

１．新規作成

kernel.res、xxx.log　ファイルが必要.

「メニュー」→「ファイル」→「新規作成」をクリック

２．開く（保存したものを開く）

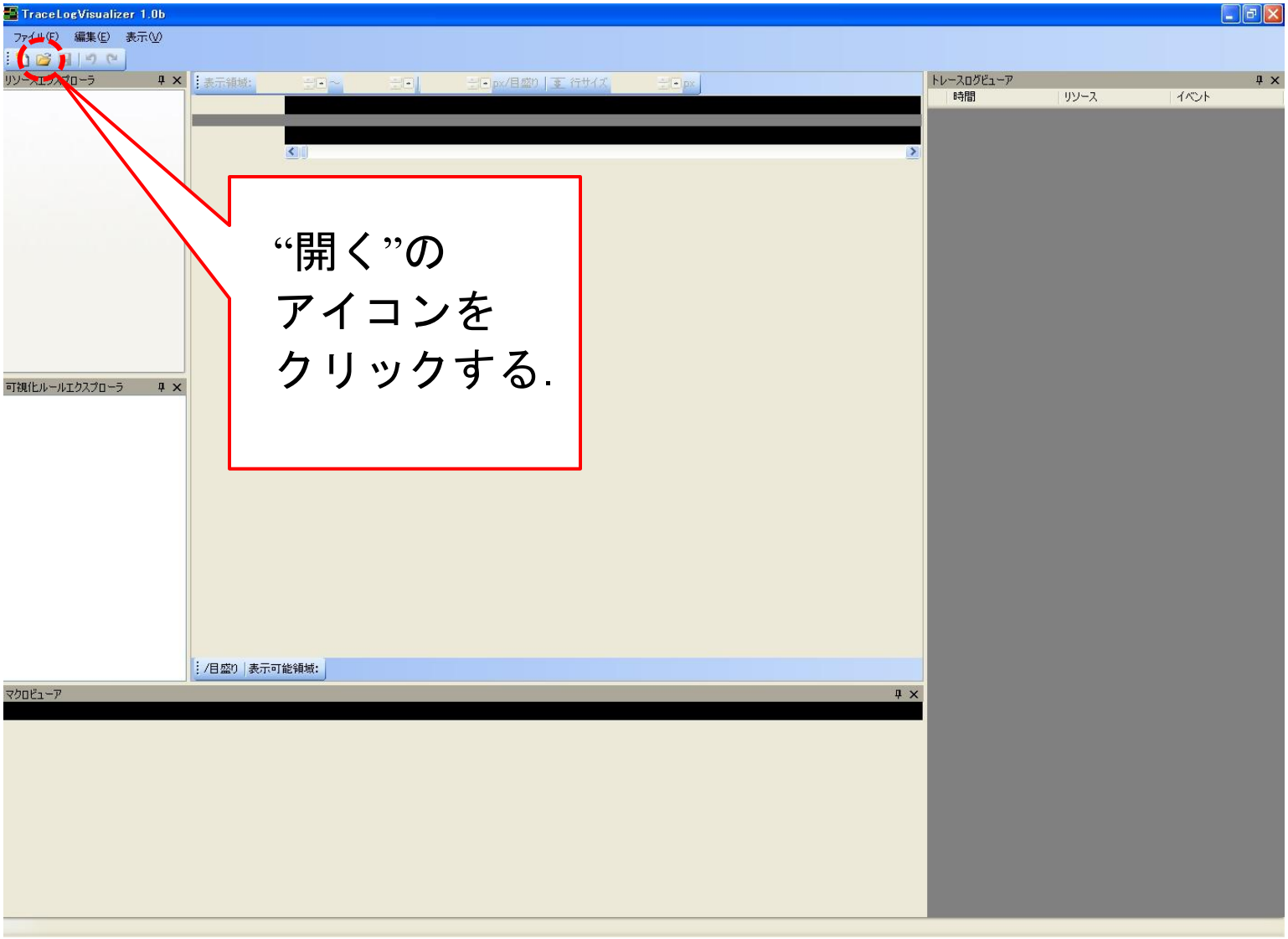
xxx.tlv　ファイルが必要.

「メニュー」→「ファイル」→「開く」をクリック

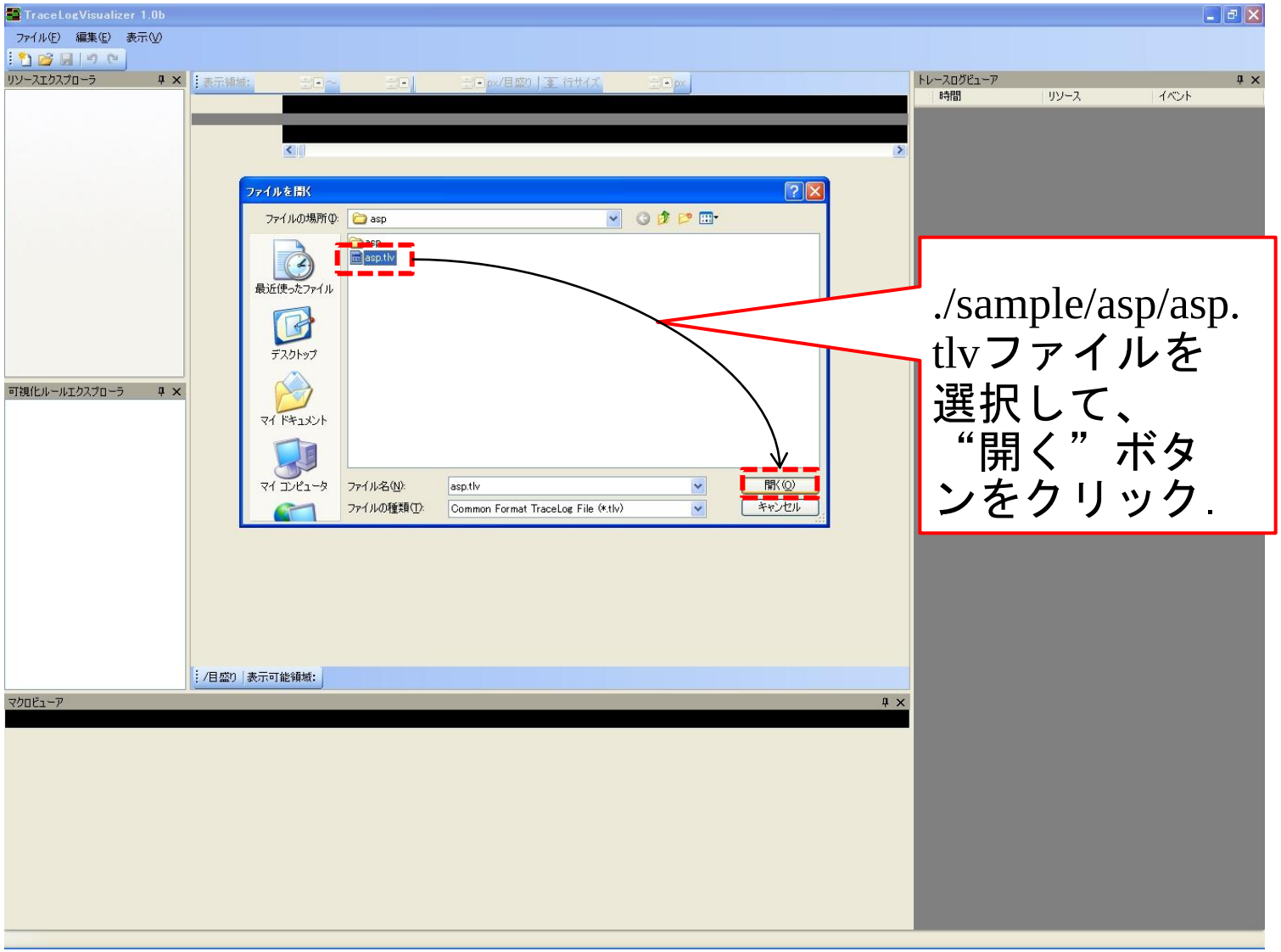
・今回は、 sampleフォルダに入っているファイルを例に説明する.

２．開く(保存したものを開く)

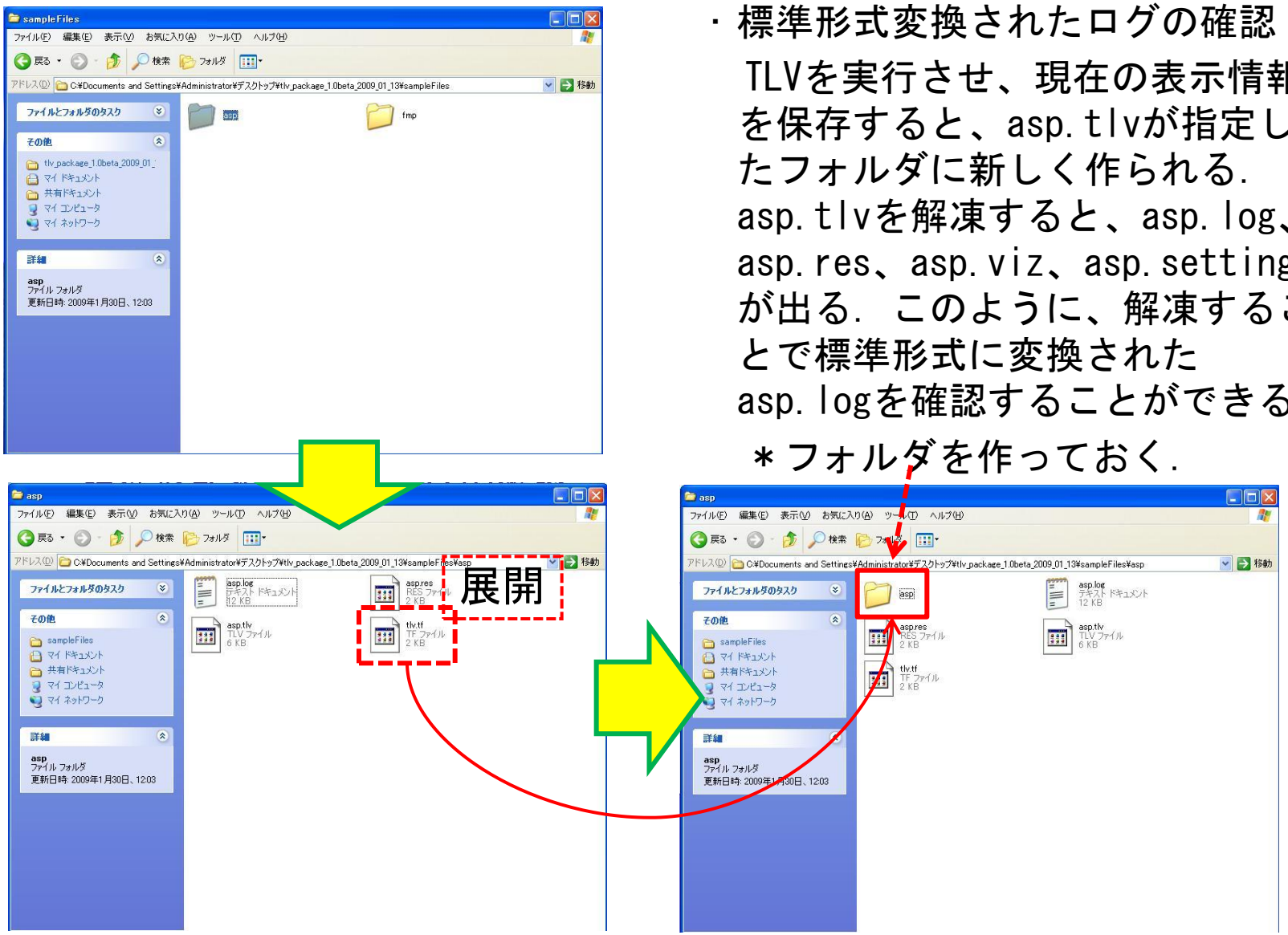
TLV初期画面



２．開く(保存したものを開く)



標準形式変換されたログの確認

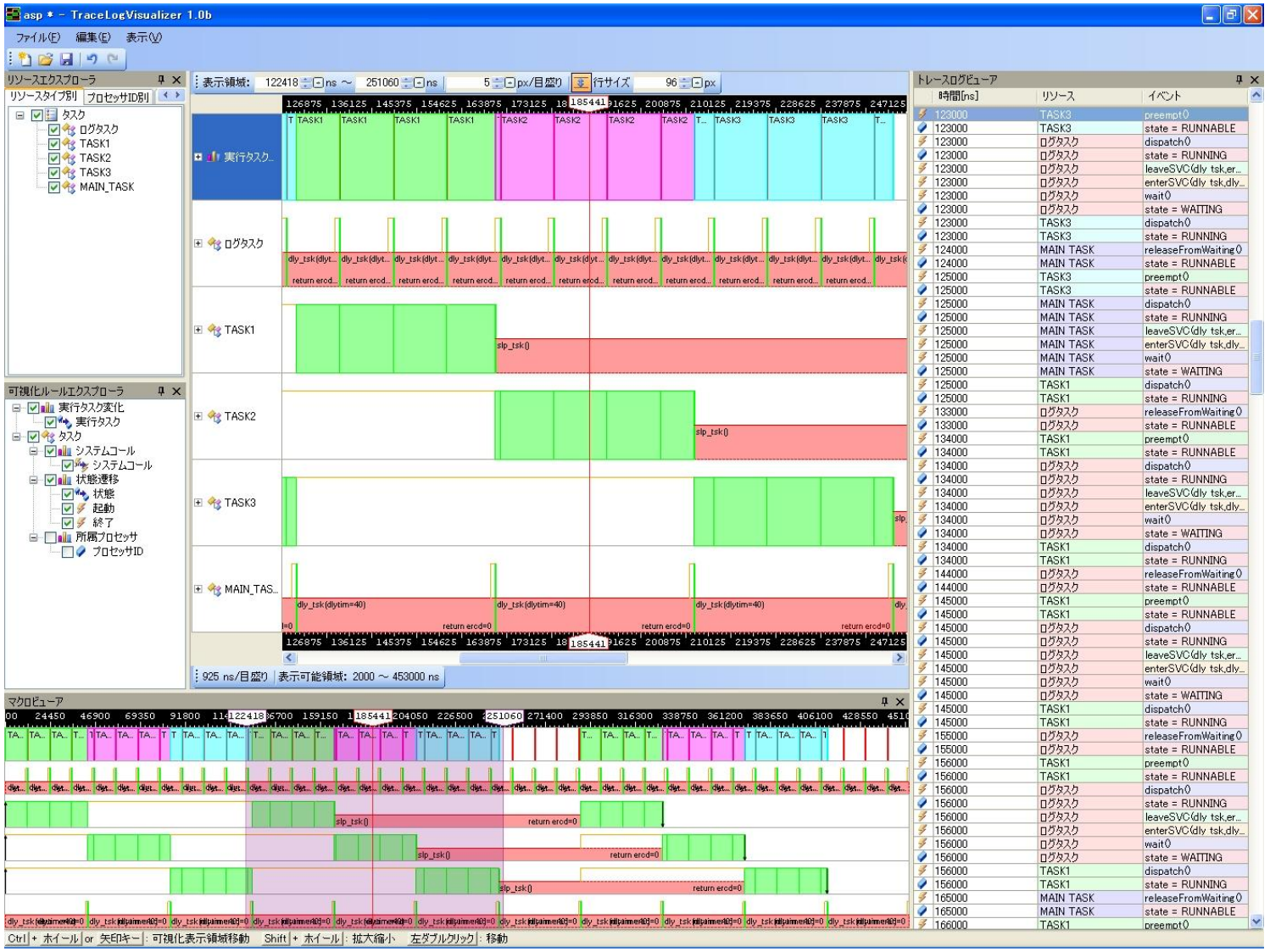


・標準形式変換されたログの確認

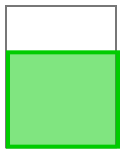
TLVを実行させ、現在の表示情報を保存すると、asp.tlvが指定したフォルダに新しく作られる.
asp.tlvを解凍すると、asp.log、asp.res、asp.viz、asp.settingが出る. このように、解凍することで標準形式に変換されたasp.logを確認することができる.

＊フォルダを作っておく.

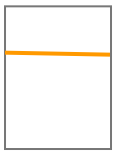
サンプルファイルを開いている画面



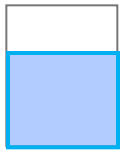
可視化表示部ー波形表示



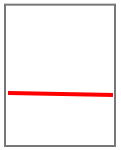
実行状態
(タスクコンテキスト)



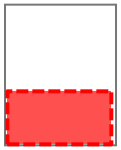
実行可能状態



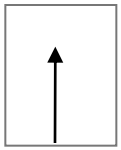
非タスクコンテキスト
(slp_tsk,dly_tsk以外)



待ち状態

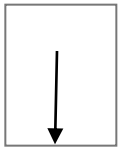


システムコール
(slp_tsk or dly_tsk)



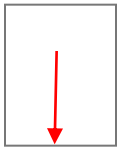
起動

システムコール
(slp_tsk,dly_tsk以外)



終了

休止状態



強制終了

TLVの各情報表示画面

TLVの各情報表示画面

1. リソースエクスプローラ ： リソース情報画面表示

2. 可視化ルールエクスプローラ ： 可視化情報画面表示

3. トレースログビュアー ： トレースログ画面表示

4. マクロビューア: マクロ画面表示

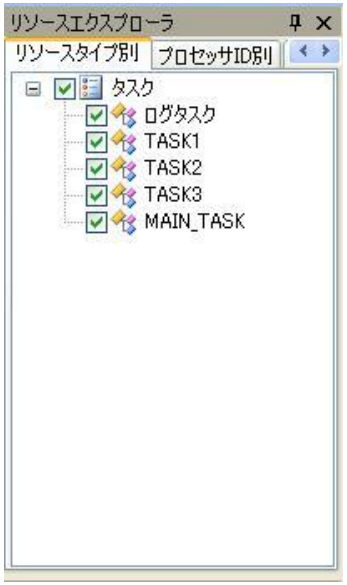
リソースエクスプローラ

・リソース情報画面表示

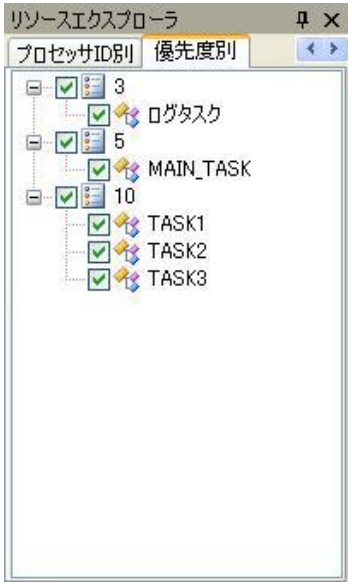
・タブをクリックして変えることにより、リソースタイプ別、優先度別にリソースを見ることができる.

・プロセッサIDはFMPの場合だけ表示

・リソースタイプ別



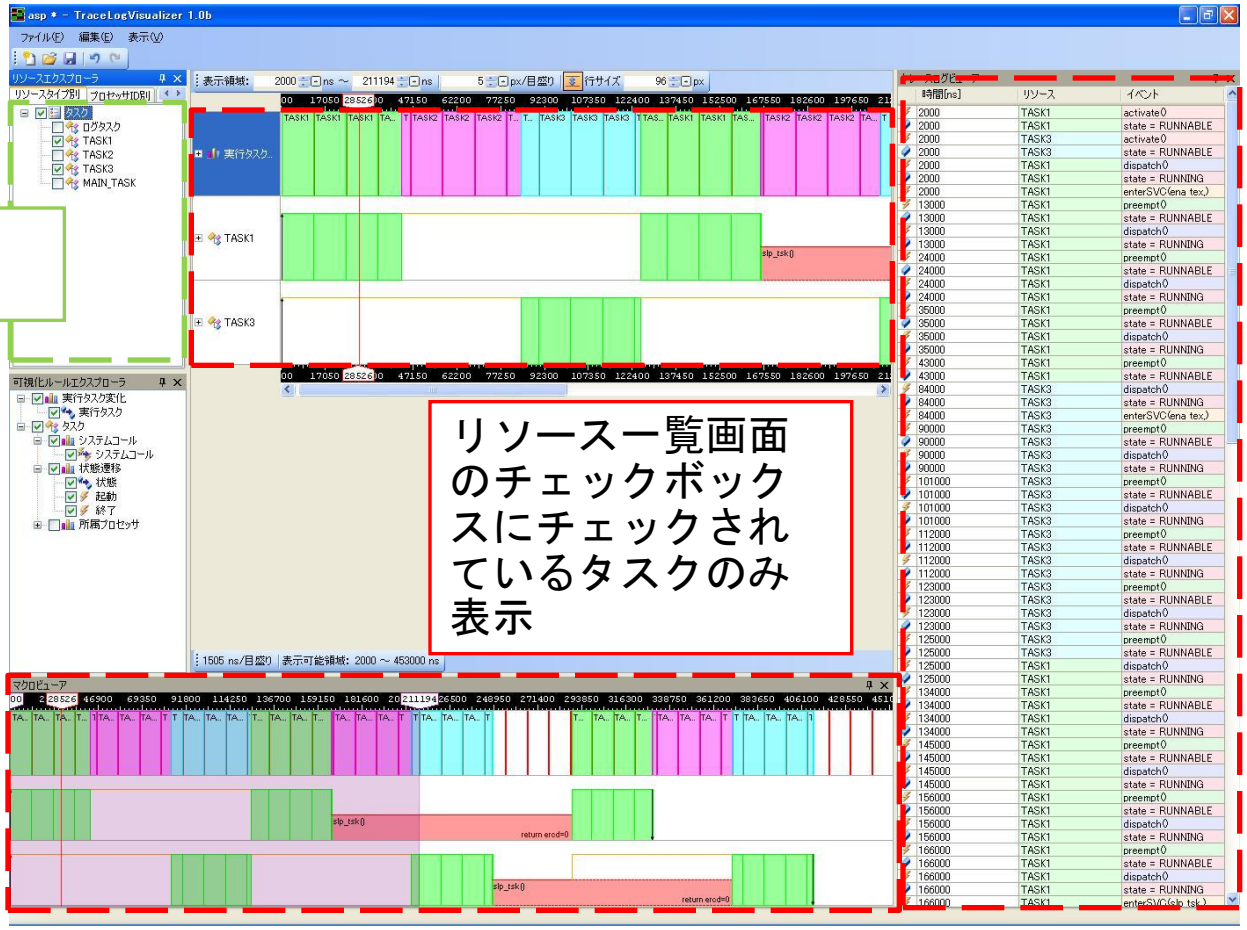
・優先度別



リソースエクスプローラ

・リソース情報画面表示部で表示するタスクのチェックボックスを選択すると可視化表示部へタスク状態が表示される

チェックボックス
をクリックする



TLVの各情報表示画面

1. リソースエクスプローラ ： リソース情報画面表示

2. 可視化ルールエクスプローラ ： 可視化情報画面表示

3. トレースログビュアー ： トレースログ画面表示

4. マクロビューア: マクロ画面表示

可視化ルールエクスプローラ

・可視化情報画面表示

・チェックボックスにチェックされている属性のみ表示.

TLVの各情報表示画面

1. リソースエクスプローラ ： リソース情報画面表示

2. 可視化ルールエクスプローラ ： 可視化情報画面表示

3. トレースログビュアー ： トレースログ画面表示

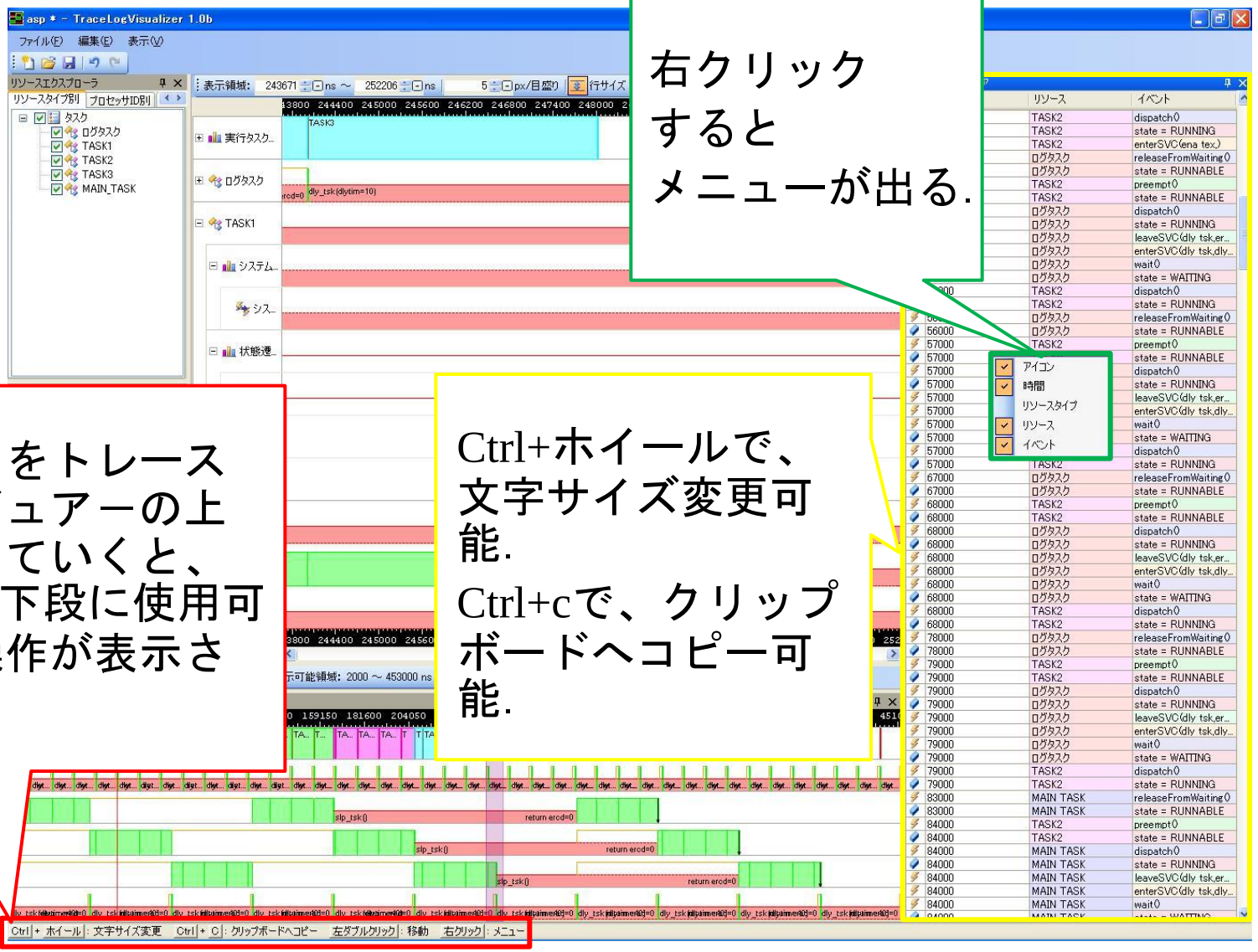
4. マクロビューア: マクロ画面表示

トレースログビューア

- トレースログ画面表示
- ソート

  時間、リソース、イベントなどのタイトル部分をクリックするとソート可能.
- 左ダブルクリックで、クリックしたところにマーカが移動.

トレースログビューア


右クリック
すると
メニューが出る.
マウスをトレース
ログビュアーの上
に持っていくと、
TLVの下段に使用可
能な操作が表示さ
れる.
Ctrl+ホイールで、
文字サイズ変更可
能.
Ctrl+cで、クリップ
ボードへコピー可
能.
asp * - TraceLogVisualizer 1.0b
ファイル(F) 編集(E) 表示(V)
Ctrl + ホイール : 文字サイズ変更
Ctrl + C : クリップボードへコピー
左ダブルクリック : 移動
右クリック : メニュー

TLVの各情報表示画面

1. リソースエクスプローラ ： リソース情報画面表示

2. 可視化ルールエクスプローラ ： 可視化情報画面表示

3. トレースログビュアー ： トレースログ画面表示

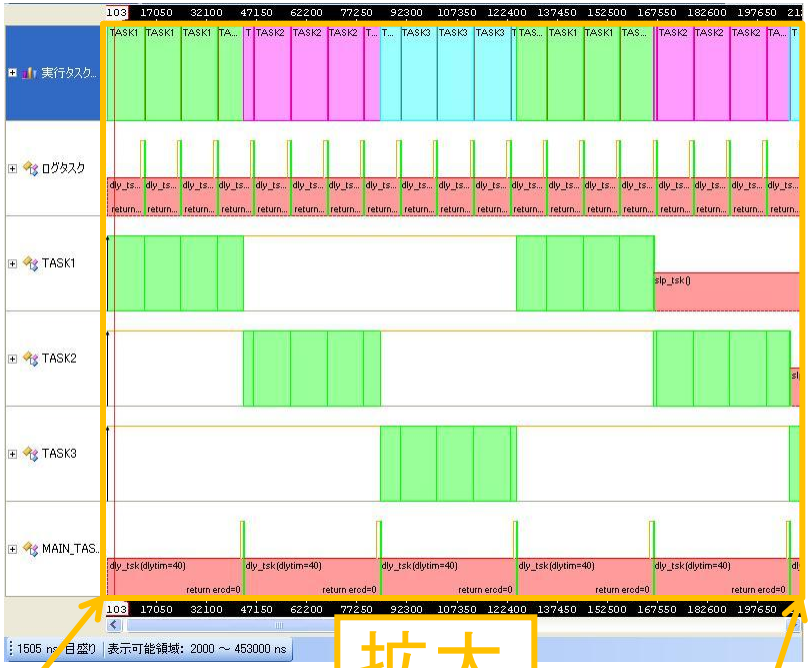
4. マクロビューア: マクロ画面表示

マクロビューアー

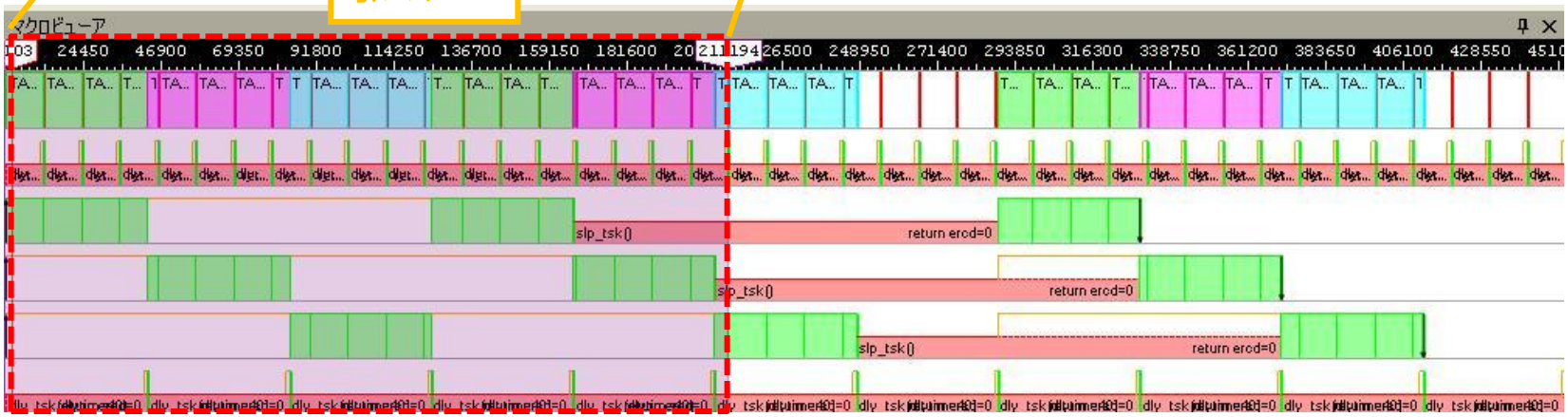
・全マクロ画面表示

・マウスでマーカを設定することにより、その部分の詳細情報が次のページに出てくる、可視化詳細表示部に表示される.

マクロビュアー－可視化詳細表示部

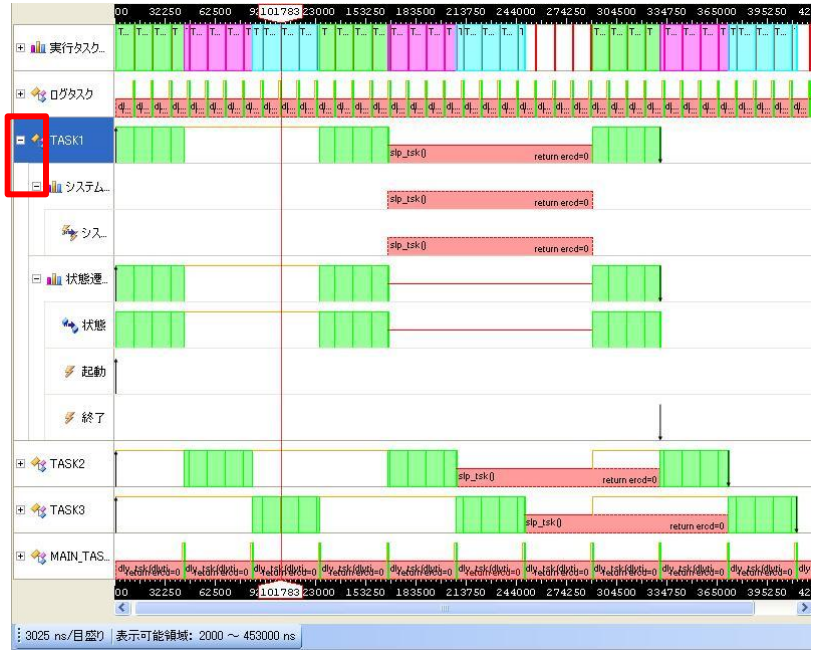


拡大



- 区間マクロ情報画面表示
- 前のマクロビュアーで、マウスでマーカを設定することにより、その部分の詳細情報が拡大されて表示される.

マクロビュアー－可視化詳細表示部



・区間マクロ情報画面表示

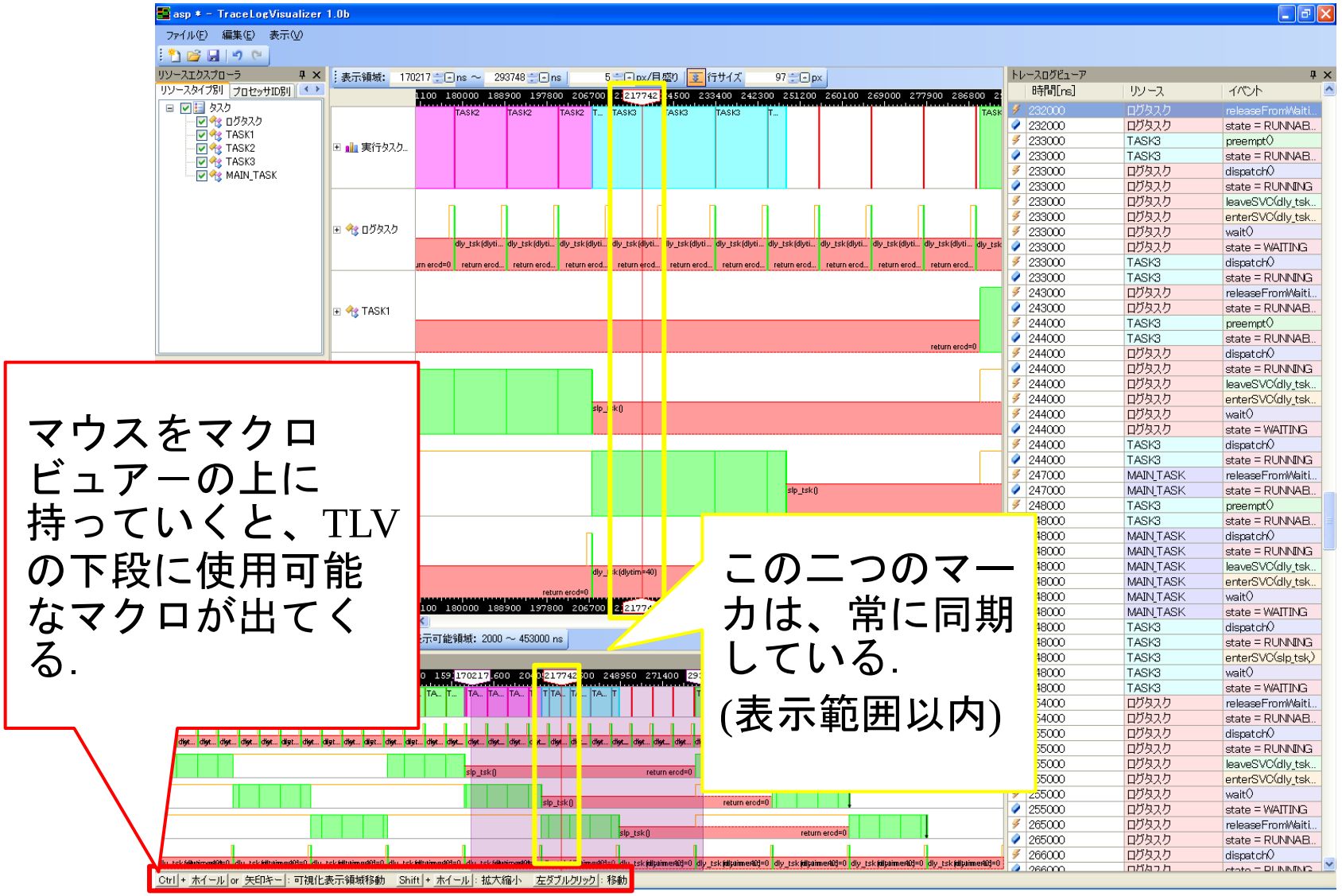
・リソースのタブを開くことによって、詳細項目ごとの振舞いを見ることができる.

・Ctrl+ホイール(矢印キー):可視化表示領域移動

・Shift+ホイール:拡大縮小

・左ダブルクリック:移動

マクロビュアー



マウスをマクロビュアーの上に持っていくと、TLVの下段に使用可能なマクロが出てくる.

この二つのマーカは、常に同期している.
(表示範囲以内)

その他の機能

- 各画面位置移動、表示、非表示
- リロード
- ツールバー
- 簡易検索

その他ー各画面位置移動、表示、非表示

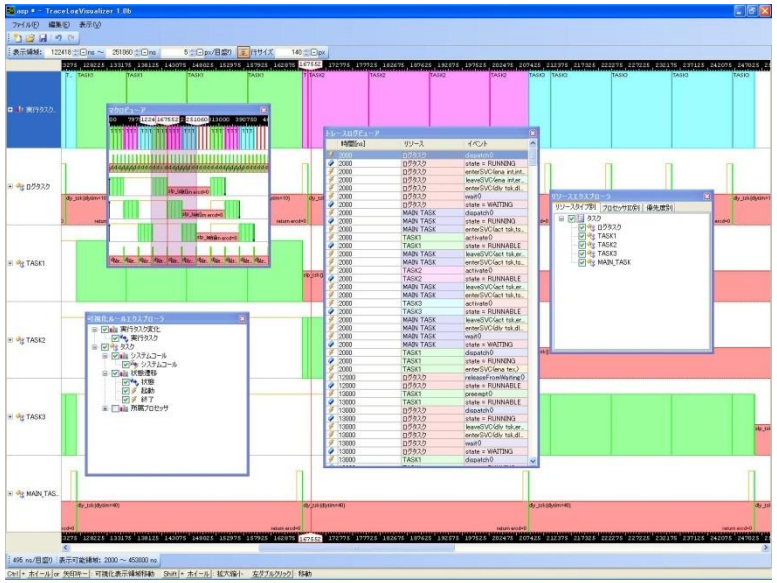
・可視化情報、リソース情報、トレースログ画面、
　マクロビューアのみ可能.

・表示は「メニュー」の「表示」でクリック.

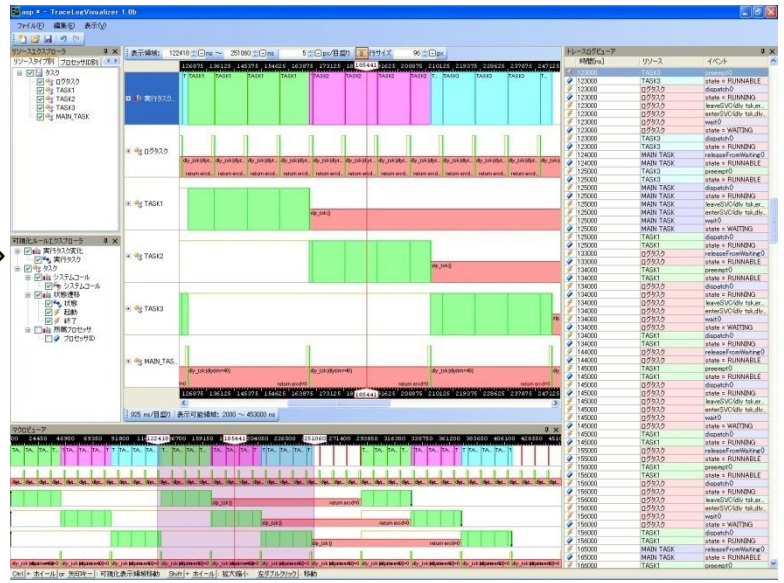
・位置移動は各画面をドラッグ＆ドロップで移動.

・windowbarをダブルクリックすると、元の位置に戻る

位置を移動した画面



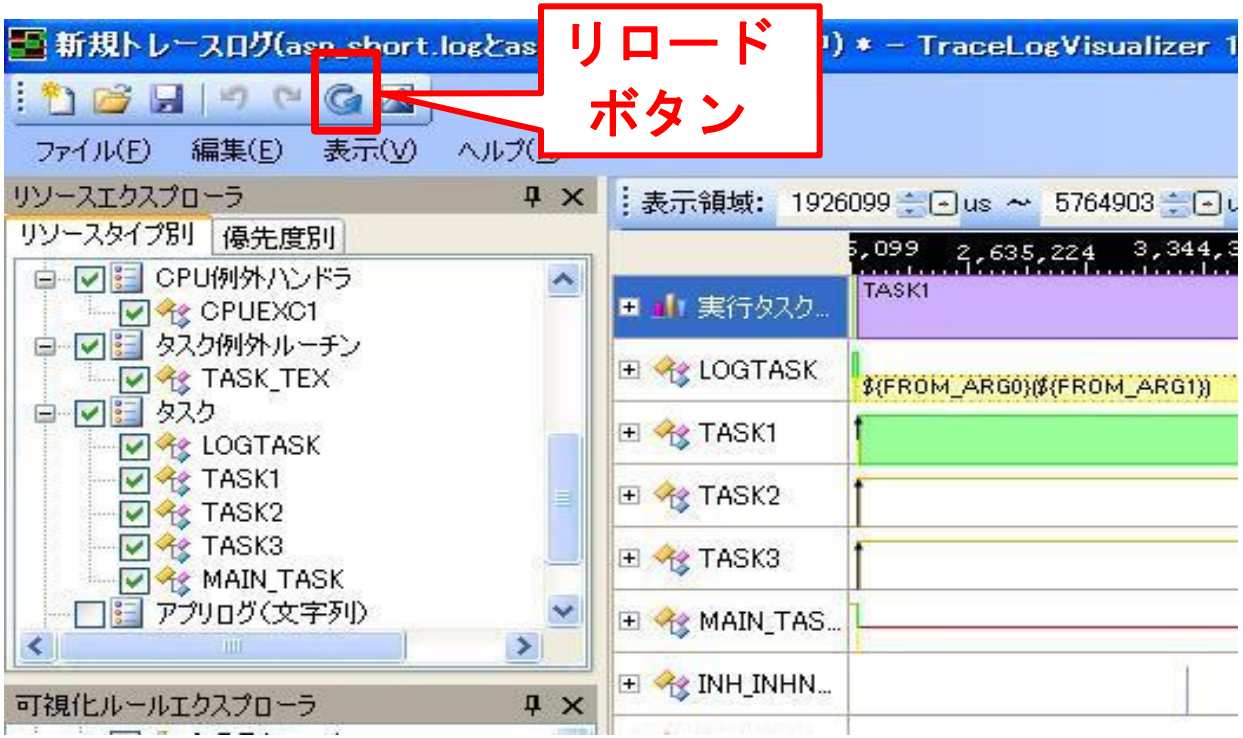
元の位置に移動した画面



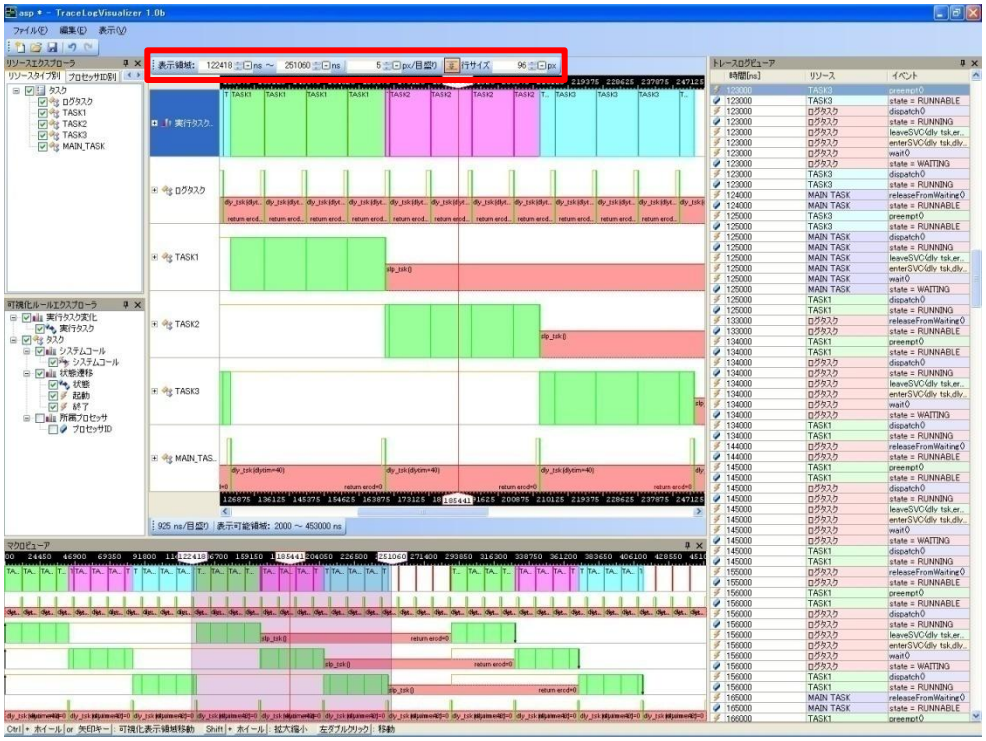
その他 ― リロード

リロード：

一度読込んだ .res ファイルと .log ファイルを再度読込みます。両ファイルをドラッグ・ドロップすることでもリロードが可能です



その他ーツールバー



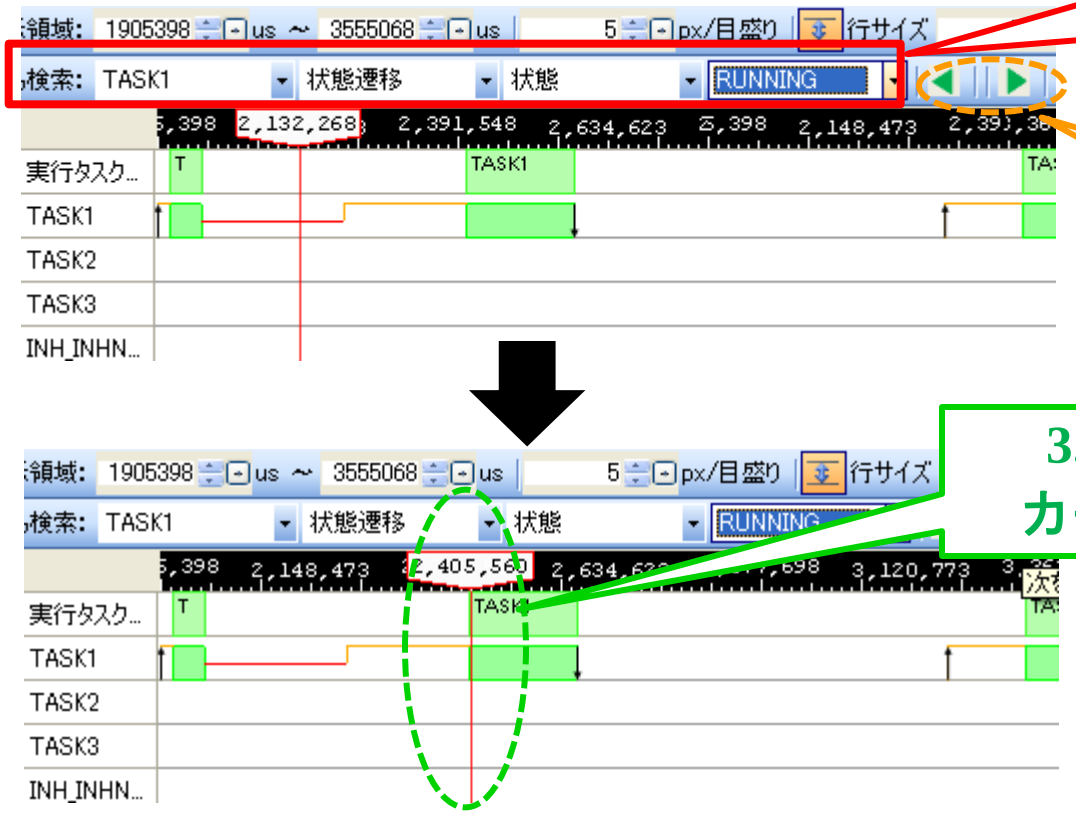
※ ツールバーの左から順に

- ns/目盛の拡大、縮小
  - ＋、ーボタンでns/目盛を拡大、縮小可能
  - 現在のns/目盛表示部分をクリックするとトラックバー(スライダー)で変更可能
- pixel/目盛の拡大、縮小
  - ＋、ーボタンでpixel/目盛を拡大、縮小可能
  - 現在のpixel/目盛表示部分をクリックするとトラックバー(スライダー)で変更可能
- 行サイズ固定
  - タスク状態表示部をウィンドウズに合わせて行サイズ固定

その他 – 簡易検索　～概要～

- **簡易検索とは**

  リソース、ルール、イベントといった検索条件を指定することで、条件に合致した時刻を探し出せる



1. 検索条件を指定（ドロップダウンリストから選択）

2. 検索ボタンを押す

3. 該当時刻へカーソルが移動

その他 – 簡易検索　～検索条件について～

- **簡易検索では以下の４つの検索条件を指定できる**

1. リソース名　　2. ルール名　　3. イベント名　　4. イベント詳細

※1 は リソースファイルで定義されている要素

2,3,4 は可視化ルールファイルで定義されている要素である

```
“taskStateChange”:{   // ルール名
   "DisplayName":"状態遷移",
   "Target":"Task",
   "Shapes":{
      “stateChangeEvent”:{    //イベント名
      "DisplayName":"状態",
      "From":"${TARGET}.state",
      "To":"${TARGET}.state",
      "Figures":{
         “${FROM_VAL}==RUNNING”       :“runningShapes”,       //イベント詳細
         "${FROM_VAL}==RUNNABLE"    :"runnableShapes“,
```

※ “TLV_1.2/visualizeRules/toppers_rules.vis”　の一部を抜粋

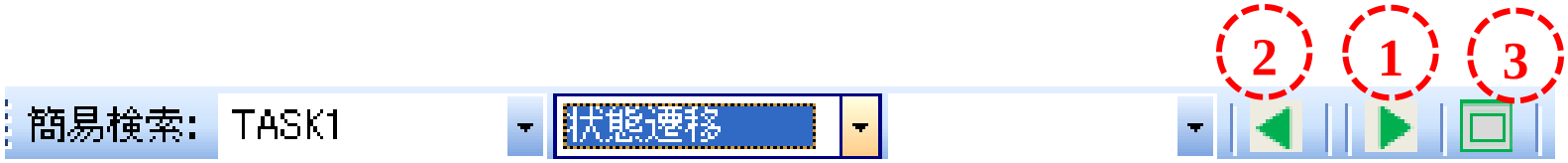
その他 – 簡易検索　～検索の種類について～

•検索の種類

1. **前方検索**
   検索条件に合致する、現在から一番近い次の時刻へ移動する

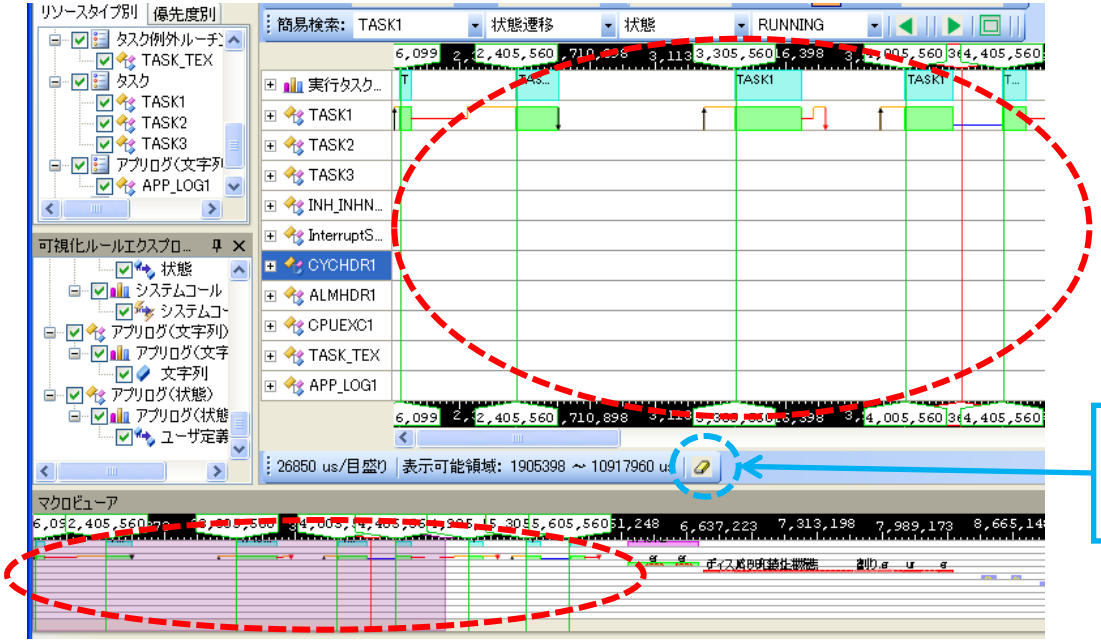
2. **後方検索**
   検索条件に合致する、現在から一番近い前の時刻へ移動する

3. **全体検索**
   検索条件に合致する全ての時刻にマーカーを引く



その他 – 簡易検索　～全体検索について～

- 全体検索では、可視化表示部分の該当箇所にマーカーが引かれます



※マーカーは全消去ボタンで全て消すことができます

※全体検索ボタンを押した際、一部表示画面にはすぐにマーカーが引かれます。
一度マウスカーソルを一部表示画面、もしくは全体表示部分に移動させると全体表示部分にもマーカーが反映されます