

ISE Project Navigator での SmartXplorer の使用

チュートリアル (ISE 12.1.1)

UG689 (v12.1.1) 2010 年 5 月 28 日



Xilinx is disclosing this user guide, manual, release note, and/or specification (the "Documentation") to you solely for use in the development of designs to operate with Xilinx hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Xilinx expressly disclaims any liability arising out of your use of the Documentation. Xilinx reserves the right, at its sole discretion, to change the Documentation without notice at any time. Xilinx assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU "AS-IS" WITH NO WARRANTY OF ANY KIND. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

© 2010 Xilinx, Inc. XILINX, the Xilinx logo, Virtex, Spartan, ISE, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

本資料は英語版 (v12.1.1) を翻訳したもので、内容に相違が生じる場合には原文を優先します。

資料によっては英語版の更新に対応していないものがあります。

日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。

目次

このチュートリアルについて	
チュートリアルの内容.....	5
その他のリソース	6
表記規則.....	6
書体.....	6
オンライン マニュアル.....	7
第 1 章： SmartXplorer の概要	
概要	9
SmartXplorer の主な利点	9
デザイン ストラテジ	9
並列実行	10
Linux OS	10
Microsoft Windows OS.....	10
1 つの Linux または Windows マシンの使用	10
第 2 章： チュートリアルの説明	
演習の実行に必要な知識.....	12
第 3 章： 演習の準備	
デザインの概要	13
手順 (Linux および Windows)	13
第 4 章： 演習 1： 基本フロー	
目標	15
演習	15
手順 1： 演習プロジェクトを開く.....	15
手順 2： デフォルトのオプションを使用した SmartXplorer の実行	15
手順 3： SmartXplorer の結果の使用	19
手順 4： 実行を追加で反復してタイミングを向上	21
手順 5： あらかじめ定義されたストラテジと追加の反復を一度に実行	23
まとめ	23
第 5 章： 演習 2： カスタム ストラテジの作成	
目標	25
演習	25
手順 1： 演習プロジェクトを開く.....	25
手順 2： 2 つの最良のストラテジの特定	25
手順 3： カスタム ストラテジ ファイルの作成	26
まとめ	28
第 6 章： 演習 3W： 複数のストラテジの同時実行 (Windows)	
目標	29
演習	29
手順 1： 演習プロジェクトを開く.....	29
手順 2： 並列実行可能なストラテジ数を 2 に設定して SmartXplorer を実行	30
まとめ	31

第 7 章：演習 3L：複数のストラテジの同時実行 (Linux)

目標	33
演習	33
手順 1：ザイリンクス環境と結果の保存.....	33
手順 2：実行前のチェックリスト.....	34
手順 3：演習プロジェクトを開く.....	34
手順 4：通常の Linux ネットワークでの SmartXplorer の実行	35
手順 5：LSF または SGE での SmartXplorer の実行	36
まとめ	37

付録 A：カスタム ファイル

目標	39
カスタム ストラテジ ファイル	39
ホスト リスト ファイル (Linux).....	40
通常の Linux ネットワーク	40
LSF コンピュータ ファーム.....	40
SGE コンピュータ ファーム	40
さまざまなタスクでの SmartXplorer の設定	41
タスク 1：あらかじめ定義されているストラテジすべてを実行	41
タスク 2：あらかじめ定義されているストラテジの最初の 3 つを実行	42
タスク 3：あらかじめ定義されているストラテジすべてを実行し、 異なるコスト テーブルを使用して追加で 5 回実行	43
タスク 4：カスタム ストラテジすべてを実行し、異なるコスト テーブルを 使用して追加で 3 回実行.....	44

このチュートリアルについて

このチュートリアルでは、SmartXplorer の概要と、その機能をタイミング クロージャを達成するために使用する方法を示します。

このチュートリアルには、2 つのバージョンがあります。

- ISE® Project Navigator ユーザー用：SmartXplorer を Project Navigator から使用する方法を説明します。
- コマンド ライン ユーザー用：ほとんどの Microsoft Windows ユーザーは ISE Project Navigator からザイリンクス ツールを使用しているので、このチュートリアルでは主に Linux プラットフォームでの SmartXplorer の使用について説明します。内容は、必要に応じて Microsoft Windows オペレーティング システムにも簡単に適用できます。

どちらのバージョンにも、詳細な手順を示す演習が含まれており、SmartXplorer のさまざまな機能を順を追って学ぶことができます。

重要：12.1 リリースから、SmartXplorer で Xilinx® Synthesis Technology (XST) および Synplify 合成ツールがサポートされるようになりました。複数のインプリメンテーション ストラテジを実行する前に、複数の合成ストラテジを実行してインプリメンテーションの実行に使用する最良の合成済みネットリストを選択できます。SmartXplorer での合成の実行は、コマンド ラインのみでサポートされ、ISE 環境では使用できません。SmartXplorer での合成の実行については、コマンド ライン ユーザー用の SmartXplorer チュートリアルを参照してください。

このチュートリアルでは、SmartXplorer のすべてのオプションおよび機能は説明していません。

チュートリアルの内容

このチュートリアルは、次の章から構成されています。

- [第 1 章「SmartXplorer の概要」](#)：SmartXplorer の機能を説明します。
- [第 2 章「チュートリアルの説明」](#)：各演習で学ぶ機能について概要を説明します。各演習を完了するのに必要な時間も示します。
- [第 3 章「演習の準備」](#)：各演習で使用するデザインの入手先とインストール方法を示します。
- 第 4、5、6、および 7 章：各演習の詳細を示します。演習の概要と、演習を完了するために必要な詳細な手順を説明します。
 - ◆ [第 4 章「演習 1：基本フロー」](#)
 - ◆ [第 5 章「演習 2：カスタム ストラテジの作成」](#)
 - ◆ [第 6 章「演習 3W：複数のストラテジの同時実行 \(Windows\)」](#)
 - ◆ [第 7 章「演習 3L：複数のストラテジの同時実行 \(Linux\)」](#)
- [付録 A「カスタム ファイル」](#)：カスタム ストラテジ ファイルおよびホスト リスト ファイルの例を示します。また、さまざまなタスクでの SmartXplorer の設定も含まれています。

その他のリソース

追加の資料は、次の Web サイトから参照できます。

<http://japan.xilinx.com/support/documentation/index.com>

シリコンやソフトウェア、IP に関するアンサー データベースを検索したり、テクニカル サポートのウェブ ケースを開く場合は、次の Web サイトにアクセスしてください。

<http://japan.xilinx.com/support>

表記規則

このマニュアルでは、次の表記規則を使用しています。各規則について、例を挙げて説明します。

書体

次の規則は、すべてのマニュアルで使用されています。

表記規則	使用箇所	例
Courier フォント	システムが表示するメッセージ、プロンプト、プログラム ファイルを表示します。	<code>speed grade: - 100</code>
Courier フォント (太字)	構文内で入力するコマンドを示します。	ngdbuild <i>design_name</i>
イタリック フォント	ユーザーが値を入力する必要のある構文内の変数に使用します。	<i>ngdbuild design_name</i>
二重/一重かぎカッコ『』、「」	『』はマニュアル名を、「」はセクション名を示します。	詳細については、『開発システムリファレンス ガイド』の「PAR」を参照してください。
角カッコ []	オプションの入力またはパラメータを示しますが、 bus[7:0] のようなバス仕様では必ず使用します。また、GUI 表記にも使用します。	ngdbuild [option_name] <i>design_name</i> [File] → [Open] をクリックします。
中カッコ { }	1 つ以上の項目を選択するためのリストを示します。	lowpwr = {on off}
縦棒	選択するリストの項目を分離します。	lowpwr = {on off}
縦の省略記号 . . .	繰り返し項目が省略されていることを示します。	IOB #1: Name = QOUT' IOB #2: Name = CLKIN' . . .
横の省略記号 ...	繰り返し項目が省略されていることを示します。	allow block <i>block_name loc1 loc2 ... locn;</i>

オンライン マニュアル

このマニュアルでは、次の規則が使用されています。

表記規則	使用箇所	例
青色の文字	マニュアル内の相互参照を示します。	詳細は、「 その他のリソース 」を参照してください。 詳細は、第 1 章「 タイトル フォーマット 」を参照してください。
青色の下線付き文字	Web サイト (URL) へのハイパーリンクです。	最新のスピード ファイルは、 http://japan.xilinx.com から入手 できます。

SmartXplorer の概要

概要

SmartXplorer は、最短時間でタイミング クロージャを達成することを目標としています。

タイミング クロージャは、FPGA デザインにおいて最大の課題の 1 つです。ザイリンクスでは、タイミング クロージャの達成をサポートするため、次の点に焦点を置いています。

- 合成およびインプリメンテーション アルゴリズムの向上
- PlanAhead[®] ソフトウェアおよび FPGA Editor などの高度なグラフィカル解析ツールの提供

FPGA ツールは使いやすくなってきており、高度な機能を提供するようになってきていますが、すべてのデザイン状況を予測するのは困難です。デザイン サイクルの最終段階になってから問題が見つかる場合もあります。

設計者の経験レベルにかかわらず、HDL コードを変更したり配置制約を使用する前に、異なるツール オプションを変更するのが通常のアプローチです。ツール オプションを変更するのは非常に簡単ですが、よりよいオプションをどのように見つけ出すかが問題となります。

SmartXplorer の主な利点

SmartXplorer には、主に次の 2 つの利点があります。

- あらかじめ定義されたインプリメンテーション ストラテジまたはユーザー定義のインプリメンテーション ストラテジを複数使用してデザインを実行し、タイミングを満たすことができます。
メモ: デザイン ストラテジとは、エリア、スピード、消費電力など、特定のデザインの目標を達成するためのプロセス プロパティとその値の組み合わせです。
- 複数のストラテジを複数のマシンで同時に実行し、ジョブを短時間で完了します。

デザイン ストラテジ

SmartXplorer では、あらかじめ定義されたストラテジが複数提供されています。これらのストラテジは、各 FPGA ファミリー用に個別に選択され、調整されています。この選択は、特定のソフトウェア バージョンで最適に機能することを確実にするため、各メジャー リリースで修正されます。

独自の経験に基づいて、カスタム ストラテジまたはスクリプトを作成することもできます。

SmartXplorer ではこれらのカスタム ストラテジをシステムに統合し、それらのみを使用したり、SmartXplorer で提供されているストラテジと組み合わせで使用できます。

SmartXplorer は、プロジェクトの最終段階だけでなく、プロジェクト サイクル全体で使用するにより、デザイン サイクルの最終段階で緊急事態が発生するのを回避または削減できます。定期的に行う実行してデザインの状態を監視し、タイミングが許容範囲内にあることを確認することをお勧めします。

並列実行

複数のデザイン ストラテジ (ジョブ) を同時に実行すると、プロジェクトを短時間で完了できます。この機能は、ご使用のオペレーティング システムによって異なります。

Linux OS

ネットワーク上の複数のマシンで複数のジョブを並列実行できます。これには、次の 2 つの方法があります。

- 通常の Linux ネットワーク：ネットワーク上でのジョブの分配は SmartXplorer で管理されます。使用するマシンのリストが必要です。
- LSF (Load Sharing Facility) または SGE (Sun Grid Engine) コンピュータ ファーム：ジョブの分配は、LSF または SGE で管理されます。SmartXplorer に同時に割り当て可能なマシンの数を指定します。

Linux ネットワークへのアクセスがない場合でも、個人の Linux マシンにマルチコア プロセッサまたは複数のプロセッサがあれば、複数のジョブを並列実行できます。

Microsoft Windows OS

1 つの Windows マシンにマルチコア プロセッサまたは複数のプロセッサが搭載されている場合、複数のストラテジを同時に実行できます。

1 つの Linux または Windows マシンの使用

ネットワーク上の Linux サーバーへのアクセスがなく、ローカル コンピュータしか使用できない場合は、マシンにマルチコア プロセッサがあるか複数のプロセッサがあることを確認してください。

まず、ご使用のマシンで同時に実行可能なジョブの数を予測する必要があります。

論理的には、同時に実行可能なジョブの数は次のように算出されます。

$$\text{Nb_Of_Jobs} = P * C$$

ここで、P はプロセッサの数、C はプロセッサごとのコアの数です。

たとえば、デュアル コア プロセッサが 4 つある場合、8 個のジョブを同時に実行できます。

ただし、使用可能なメモリ、メモリの速度、ハード ドライブの速度によっては、コンピュータで上記の式で求めた最大数のジョブを処理できない可能性があります。

次に、並列実行に関するヒントを示します。

ヒント 1：デザインのメモリ要件のため、マシンで一度に 1 つのストラテジしか実行できない場合は、すべてのストラテジを順次に行う必要があります。このような場合、夜間に SmartXplorer を実行すると有益です。

ヒント 2：タイミングの問題を解決している場合、デザインに含まれるブロックを個別に実行することもできます。これらのブロックに対してであれば複数のストラテジを同時に実行できる可能性があります、時間を節約できます。

チュートリアルの説明

このチュートリアルでは小型のデザインを使用しており、演習を短時間で完了できるようになっています。SmartXplorer の主な機能をすべて含むチュートリアル全体を、30 分以内で完了できます。

演習は、順番に実行することをお勧めします (演習 1、演習 2 など)。ただし、各演習は独立しており、特定の機能をすぐに学ぶ必要がある場合は、どの順序で実行しても問題ありません。

SmartXplorer には、主に次の 2 つの利点があります。

- あらかじめ定義されたデザイン ストラテジまたはユーザー定義のストラテジを使用することにより、タイミング クロージャの達成をサポートします。
- 複数のストラテジを複数のマシンで同時に実行し、ジョブを短時間で完了します。

このチュートリアルでは、これらの 2 つの機能を別々に説明します。

- 演習 1 および演習 2 では、タイミング クロージャについて説明します。これら 2 つの演習では、すべてのデザイン ストラテジを 1 つずつ順番に実行します (複数のストラテジを並列実行する機能はオフ)。
- 演習 3 では、複数のストラテジの並列実行について説明します。

複数のマシン間でのジョブの分配は Linux OS でのみサポートされているので、演習 3 は 2 つの部分に分かれています。

- ◆ 演習 3W は、マルチコア プロセッサまたは複数のプロセッサを持つ Windows マシン用です。
- ◆ 演習 3L は、Linux ネットワーク、LSF および SGE コンピュータ ファーム、マルチコア プロセッサまたは複数のプロセッサを持つマシン用です。

次の表に、すべての演習の概要を示します。

表 2-1：演習の概要

演習番号とタイトル	時間	説明する機能
演習 1：基本フロー	15 分	<ul style="list-style-type: none"> SmartXplorer の Project Navigator への統合についてと実行方法 最終結果のレポート方法、保存方法、および活用方法 前回の実行で得られた結果をさらに向上する方法 あらかじめ定義されたストラテジと追加の反復を一度に実行するよう SmartXplorer を設定する方法 (夜間実行など)
演習 2：カスタム ストラテジの作成	5 分	<ul style="list-style-type: none"> カスタム ストラテジ ファイルを作成し、SmartXplorer で使用する的方法
演習 3W：複数のストラテジの同時実行 (Windows)	5 分	<ul style="list-style-type: none"> 複数のストラテジの並列実行
演習 3L：複数のストラテジの同時実行 (Linux)	10 分	<ul style="list-style-type: none"> 環境を設定して複数のストラテジを並列実行する方法 通常の Linux ネットワーク、LSF および SGE コンピュータ ファーム

演習の実行に必要な知識

演習を実行するには、ISE® Project Navigator 環境に関する基本的な知識が必要です。演習を開始する前に、次の操作を実行できることが必要です。

- ザイリンクス FPGA インプリメンテーション フローの主要なステップとその実行方法
合成、変換 (NGDBuild)、マップ、配置配線、およびタイミング解析 (TRCE)
- 既存のプロジェクトを開く、閉じる

演習の準備

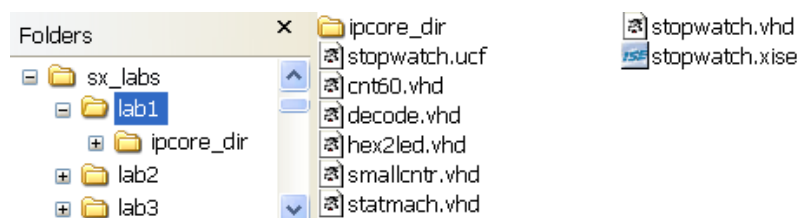
この章では、各演習用にデザインを準備する方法を説明します。

デザインの概要

このチュートリアルすべての演習で、Spartan®-6 xc6slx4-3tqg144 デバイスをターゲットとする小型の stopwatch デザインを使用します。

手順 (Linux および Windows)

1. 次のサイトから UG689.zip ファイルをダウンロードします。
<https://secure.xilinx.com/webreg/clickthrough.do?cid=148526&license=RefDesLicense>
2. SmartXplorer 演習データを保存する sx_labs ディレクトリを作成します。
3. UG689.zip ファイルを sx_labs ディレクトリにコピーします。
4. UG689.zip ファイルを解凍します。ディレクトリ構造は、次のようになります。



各演習のディレクトリには、watchvhd.xise という名前の ISE プロジェクトが含まれています。

演習 1：基本フロー

目標

この演習では、次の内容を学びます。

- SmartXplorer の Project Navigator への統合について
- あらかじめ定義されている SmartXplorer ストラテジの実行方法
- 最終結果のレポート方法、保存方法、および活用方法
- 前回の実行で得られた結果に基づき、実行を追加で反復してタイミングを向上する方法

演習

手順 1：演習プロジェクトを開く

lab1 ディレクトリにある stopwatch.xise プロジェクトを開きます。

手順 2：デフォルトのオプションを使用した SmartXplorer の実行

SmartXplorer をデフォルトのオプションを使用して実行します。フローの主な手順を示し、結果について説明します。

1. [Tools] → [SmartXplorer] → [Launch SmartXplorer] をクリックするか、またはツールバーの [Launch SmartXplorer] ボタン  をクリックします。

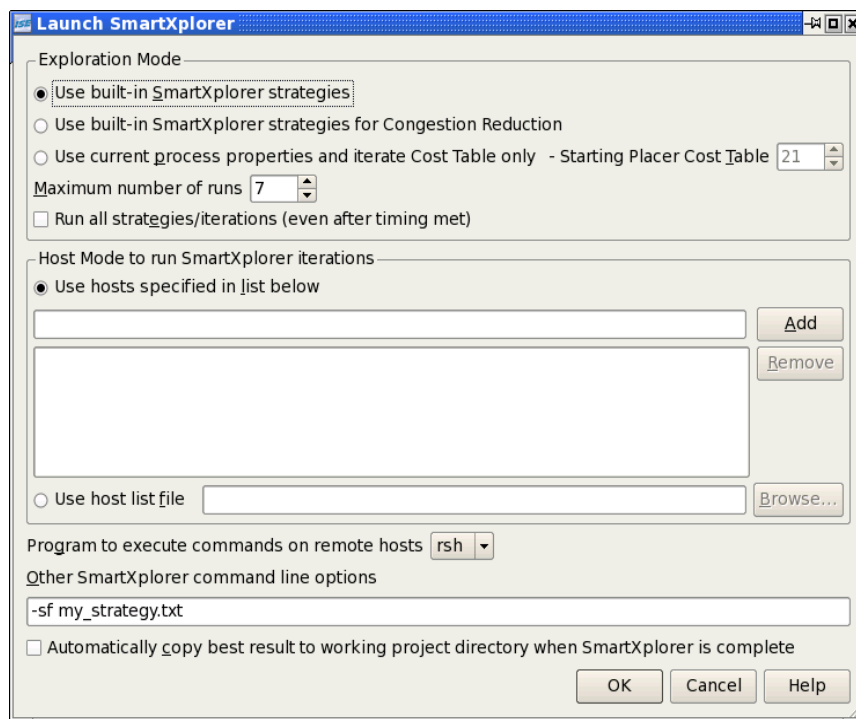


図 4-1：[Launch SmartXplorer] ダイアログ ボックス (Linux)

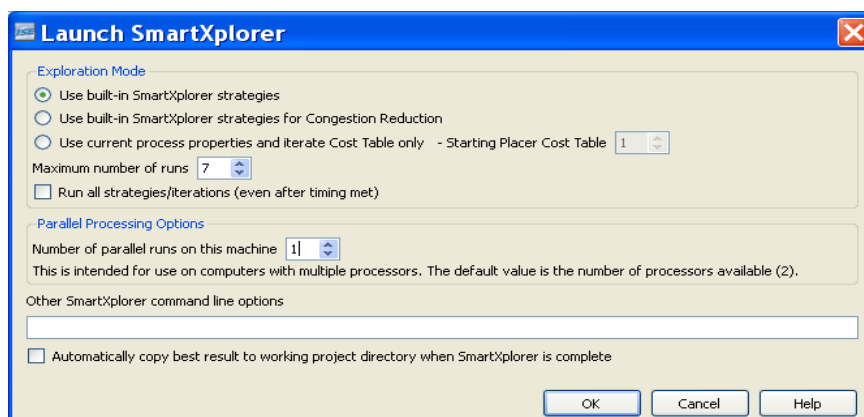
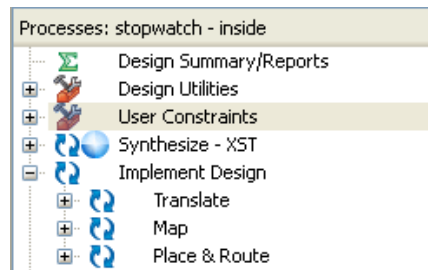


図 4-2：[Launch SmartXplorer] ダイアログ ボックス (Windows)

2. [OK] をクリックし、デフォルトのオプションで SmartXplorer を起動します。

SmartXplorer フローは、異なるオプションを使用したマップ、配置配線、TRCE の複数の実行で構成されています。各実行の最初に、必要に応じて Project Navigator で合成および変換プロセスが実行されます。



SmartXplorer の実行を開始すると、ステータスと最終的な結果を示す表が表示されます。表の各行は、Spartan-6 ファミリー用の 7 つのビルトイン ストラテジの 1 つを示します。

この表は、SmartXplorer の実行中随時アップデートされます。次に、中間状態の例を示します。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffortIOReg	None	None	None	0	None	Copy Results
MapGlobOptLogOptRegDup	local-pc	run2	Mapping	0	00h 00m 03s	Copy Results
MapRegDup	None	None	None	0	None	Copy Results
MapRunTime	local-pc	run1	Done	1806	00h 00m 40s	Copy Results
MapGlobOptIOReg	None	None	None	0	None	Copy Results
MapLogOptRegDup	None	None	None	0	None	Copy Results
MapExtraEffort2	None	None	None	0	None	Copy Results

MapRunTime ストラテジの [Status] 列は [Done] となっており、完了していることがわかります。このストラテジの実行には 40 秒かかり、タイミング スコアは 1806 であるのでタイミング制約は満たされていません。この行は緑色で表示されており、現時点ではこのストラテジで最良のタイミングが得られていることを示します。

MapGlobOptLogOptRegDup ストラテジは現在実行中で、マップ段階です。

SmartXplorer ですべてのストラテジの実行が完了すると、表は次のようになります。

MapGlobOptLogOptRegDup ストラテジのタイミング スコアが最小であり、タイミングの点からこれが最良のストラテジであることがわかります。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffortIOReg	local-pc	run5	Done	6785	00h 00m 39s	Copy Results
MapGlobOptLogOptRegDup	local-pc	run2	Done	248	00h 00m 44s	Copy Results
MapRegDup	local-pc	run4	Done	1081	00h 00m 38s	Copy Results
MapRunTime	local-pc	run1	Done	1806	00h 00m 40s	Copy Results
MapGlobOptIOReg	local-pc	run3	Done	7690	00h 00m 42s	Copy Results
MapLogOptRegDup	local-pc	run6	Done	1612	00h 00m 39s	Copy Results
MapExtraEffort2	local-pc	run7	Done	1817	00h 00m 39s	Copy Results

3. MapGlobOptLogOptRegDup ストラテジの [Status] 列の [Done] リンクをクリックするとデザイン サマリが開き、ここから MapGlobOptLogOptRegDup ストラテジに対して生成されたすべてのレポートを参照できます。

Performance Summary

Final Timing Score:	248 (Setup: 248, Hold: 0, Component Switching Limit: 0)	Pinout Data:	Pinout Report
Routing Results:	All Signals Completely Routed	Clock Data:	Clock Report
Timing Constraints:	X 1 Failing Constraint		

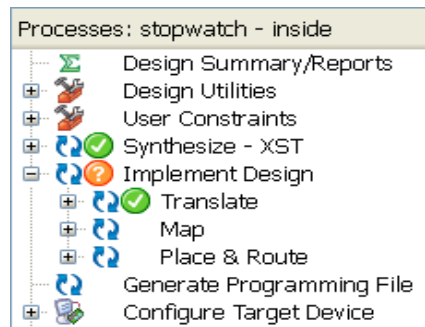
Detailed Reports (working directory="C:/sx_tutorial/lab1/smartexplorer_results/run2")

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Tue May 4 15:31:42 2010	0	1 Warning (1 new)	2 Infos (2 new)
Translation Report	Current	Tue May 4 15:31:46 2010	0	0	0
Map Report	Current	Tue May 4 15:33:05 2010	0	0	8 Infos (8 new)
Place and Route Report	Current	Tue May 4 15:33:14 2010	0	1 Warning (1 new)	0
Power Report					
Post-PAR Static Timing Report	Current	Tue May 4 15:33:20 2010	0	0	2 Infos (2 new)

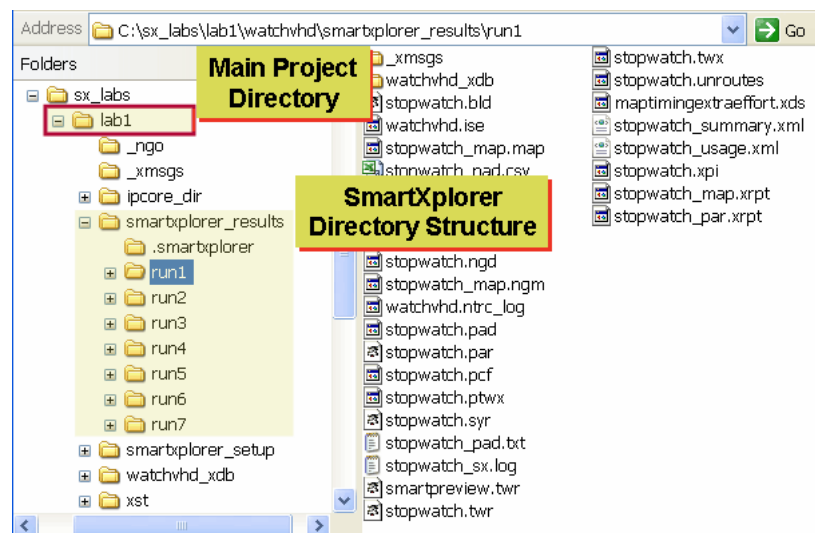
手順 3 : SmartXplorer の結果の使用

生成された SmartXplorer の結果を使用する方法を説明します。

[Design] パネルの [Processes] ペインでは、[Map] および [Place & Route] プロセスにステータスアイコンがなく、実行されていないことが示されています。これは、SmartXplorer フローが ISE Project Navigator 環境に通常のフローと並列に存在しているからです。

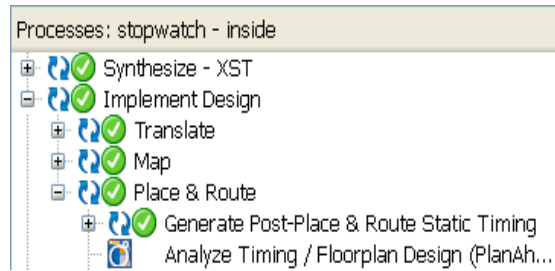


SmartXplorer のすべての結果は、通常のフローとは別に smartxplorer_results ディレクトリに保存されます。各 SmartXplorer ストラテジの結果は、個別のディレクトリ (run1、run2、...、run7) に保存されます。

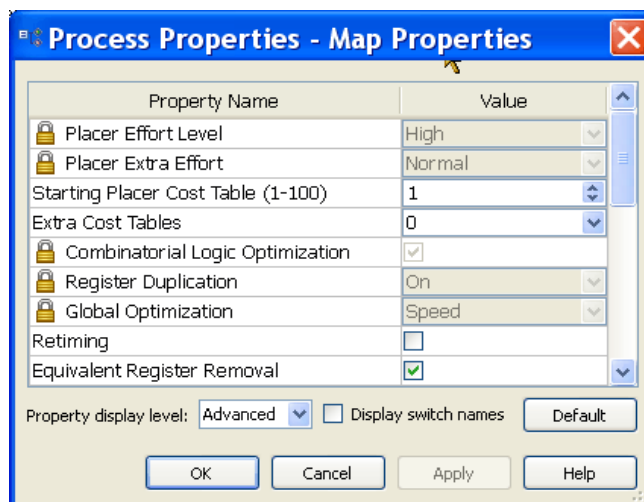


SmartXplorer の結果を、今後の実行で使えるよう通常の Project Navigator フローにインポート (コピー) できます。結果をコピーすると、プログラム ファイルの生成、詳細なタイミング解析、シミュレーション、消費電力予測を実行できます。結果をコピーするには、次の手順に従います。

1. MapGlobOptLogOptRegDup ストラテジの結果を通常の Project Navigator フローにコピーするには、SmartXplorer の結果の表で [Copy Results] をクリックします。Project Navigator で、次の処理が実行されます。
 - ◆ マップ、配置配線、およびタイミング解析のすべてのファイルが watchvhd プロジェクトディレクトリにコピーされます。通常のフローのデザイン サマリもアップデートされます。
 - ◆ マップ、配置配線、およびタイミング解析プロセスのステータスがアップデートされます。



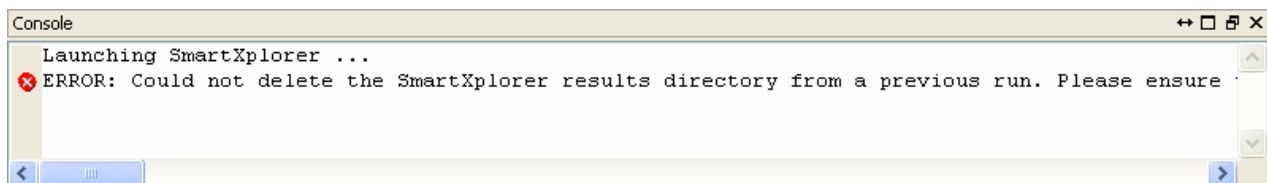
- ◆ 現在のマップおよび配置配線オプションが MapGlobOptLogOptRegDup ストラテジのオプションに置き換えられます。ストラテジで明示的に指定されているオプションはすべてロックされ、[Process Properties] ダイアログ ボックスでプロパティの横に鍵アイコンが表示されます。



メモ：ロックされているオプションを変更する場合は、[Project] → [Design Goals & Strategies] をクリックしてロックを解除する必要があります。

2. SmartXplorer の結果を活用して、通常の Project Navigator フローで設計を継続します。

重要： SmartXplorer を再実行するには、SmartXplorer の結果に対して開いているデザイン サマリおよびレポートをすべて閉じる必要があります。これは、SmartXplorer が起動する前に以前に生成されたデータ ファイルがすべて削除されるからです。開いているレポートがあると、SmartXplorer で以前に生成された結果をクリーンアップできません。この場合、[Console] パネルに「permission denied」というエラーが表示され、SmartXplorer が停止します。



手順 4 : 実行を追加で反復してタイミングを向上

前の手順で、MapGlobOptLogOptRegDup ストラテジで最良のタイミング結果が得られました。このストラテジを異なるコスト テーブルを使用してさらに 5 回実行し、タイミングを向上します。コスト テーブルの詳細は、『コマンド ライン ツール ユーザー ガイド』を参照してください。

メモ： この手順を実行する前に、MapGlobOptLogOptRegDup ストラテジの結果を通常の Project Navigator 環境にインポートする必要があります。この演習では、この操作は「[手順 3 : SmartXplorer の結果の使用](#)」で実行済みです。

1. [Tools] → [SmartXplorer] → [Launch SmartXplorer] をクリックし、[Launch SmartXplorer] ダイアログ ボックスで [Use the current process properties and iterate Cost Table only] をオンにします。
2. 前の手順で、MapGlobOptLogOptRegDup ストラテジはデフォルトのコスト テーブル 1 で実行されているので、[Starting Placer Cost Table] を 2 に設定します。これにより、追加の実行ではコスト テーブル 2、3... が使用されます。
3. [Maximum number of runs] を 5 に設定します。

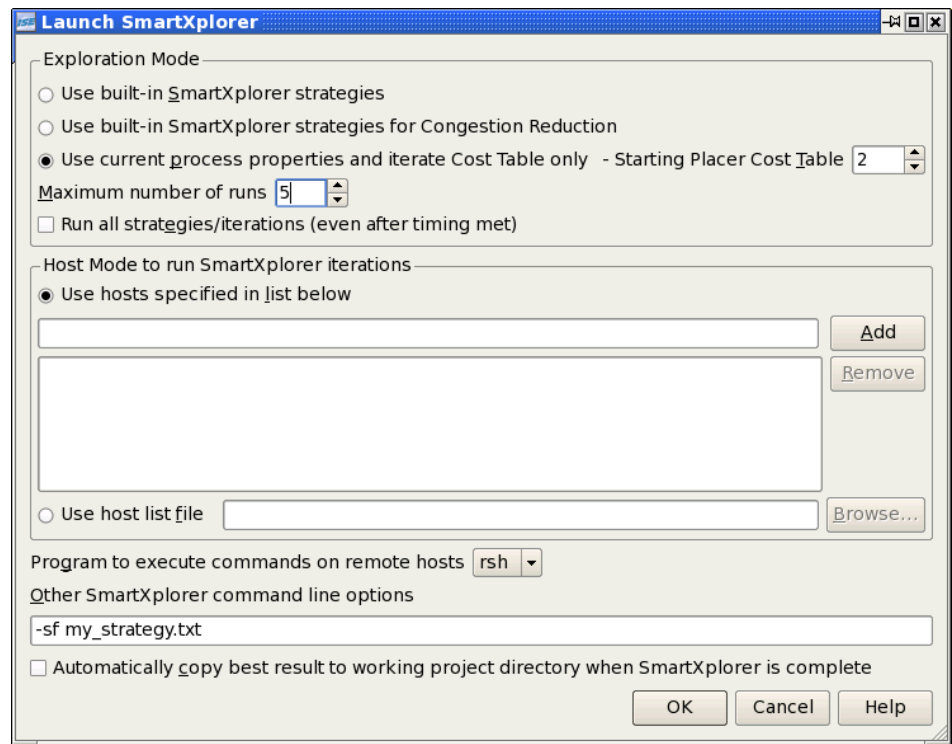


図 4-3 : コスト テーブルと実行回数の設定 (Linux)

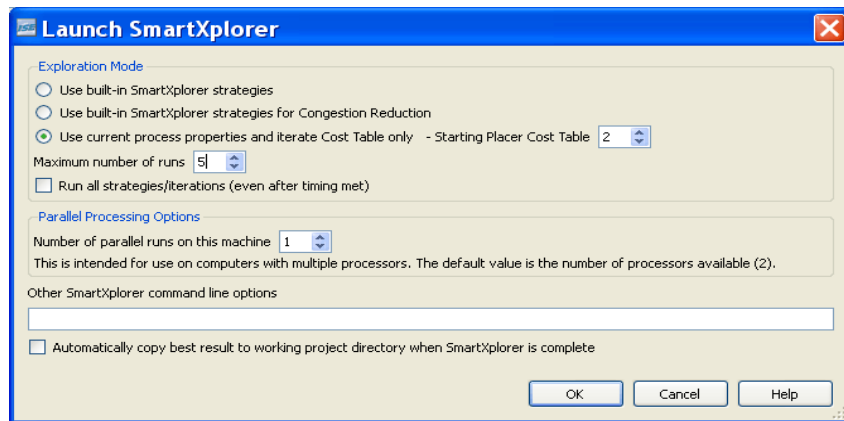


図 4-4：コスト テーブルと実行回数の設定 (Windows)

4. [OK] をクリックします。

SmartXplorer の追加実行により、タイミングが満たされました。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
CurrentProjectNavigatorSettings	local-pc	run1	Done	492	00h 00m 44s	Copy Results
CurrentProjectNavigatorSettingsCT3	local-pc	run2	Done	0	00h 00m 44s	Copy Results

ストラテジは 2 回しか実行されませんでした。これは、タイミング制約が満たされた時点で SmartXplorer の実行を終了するようデフォルトで設定されているからです。コスト テーブル 3 (MapGlobOptLogOptRegDupCT3) でタイミングが満たされたため、その時点で SmartXplorer の実行が終了しました。

[Launch SmartXplorer] ダイアログ ボックスで [Run all strategies/iterations] をオンにすると、タイミングが満たされた場合でも指定回数だけストラテジが実行され、各コスト テーブルのパフォーマンスを比較できます。

メモ：この手順で説明した手法は、異なる方法で使用できます。たとえば、通常の Project Navigator フローを使用していて十分小さいタイミング スコアが得られた場合、SmartXplorer を使用して現在のプロジェクト設定で異なるコスト テーブルを適用できます。

手順 5：あらかじめ定義されたストラテジと追加の反復を一度に実行

前の手順では、2 段階でタイミング制約を満たすことができました。最初の段階「[手順 3：SmartXplorer の結果の使用](#)」であらかじめ定義されたストラテジをすべて実行して最良のストラテジを見つけ、次の段階「[手順 4：実行を追加で反復してタイミングを向上](#)」で最良のストラテジを異なるコスト テーブルを使用して実行しました。

同じ手順を、1 つの段階で実行できます。これは、夜間にジョブを実行する場合などに有益です。**[Launch SmartXplorer]** ダイアログ ボックスで、次のように設定します。

1. **[Use built-in SmartXplorer strategies]** をオンにします。
2. **[Maximum number of runs]** を 12 (あらかじめ定義されたストラテジ 7 つ + 5 つの追加実行) に設定します。
3. タイミングが満たされた場合でもすべてのストラテジを実行する場合は、**[Run all strategies/ iterations (even after timing met)]** をオンにします。
4. **[OK]** をクリックします。

[SmartXplorer] ウィンドウに、次のような結果が表示されます。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffort2	local-pc	run7	Done	1817	00h 00m 39s	Copy Results
MapExtraEffortIOReg	local-pc	run5	Done	6785	00h 00m 39s	Copy Results
MapGlobOptIOReg	local-pc	run3	Done	7690	00h 00m 45s	Copy Results
MapGlobOptLogOptRegDup	local-pc	run2	Done	248	00h 00m 44s	Copy Results
MapGlobOptLogOptRegDupCT2	local-pc	run8	Done	492	00h 00m 44s	Copy Results
MapGlobOptLogOptRegDupCT3	local-pc	run9	Done	0	00h 00m 44s	Copy Results
MapGlobOptLogOptRegDupCT4	local-pc	run10	Done	0	00h 00m 44s	Copy Results
MapGlobOptLogOptRegDupCT5	local-pc	run11	Done	0	00h 00m 44s	Copy Results
MapGlobOptLogOptRegDupCT6	local-pc	run12	Done	344	00h 00m 44s	Copy Results
MapLogOptRegDup	local-pc	run6	Done	1612	00h 00m 39s	Copy Results
MapRegDup	local-pc	run4	Done	1081	00h 00m 39s	Copy Results
MapRunTime	local-pc	run1	Done	1806	00h 00m 39s	Copy Results

まとめ

この演習では、SmartXplorer を Project Navigator 環境から実行し、あらかじめ定義されている 7 つのストラテジを実行して結果を取得しました (段階 1「[手順 3：SmartXplorer の結果の使用](#)」)。

その後、今後の実行で使用するため SmartXplorer の結果を通常の Project Navigator フローにインポートしました。

また、最良のストラテジを 5 つの異なるコスト テーブルを使用して実行し、タイミングを向上しました (段階 2「[手順 4：実行を追加で反復してタイミングを向上](#)」)。

最後に、あらかじめ定義されているストラテジと追加の反復を一度に実行する方法を説明しました。

演習 2：カスタム ストラテジの作成

目標

あらかじめ定義されている 7 つのストラテジを実行した結果、2 つのストラテジでほかの 5 つのストラテジより常により結果が得られた場合、今後の実行ではこれら 2 つのストラテジのみを実行し、ほかの 5 つはスキップします。

また、デザインで最適な結果を得られるストラテジを手動で見つけた場合、これらのストラテジを SmartXplorer で実行できると便利です。

どちらの場合でも、カスタム ストラテジ ファイルを作成し、SmartXplorer で使用できます。

この演習では、カスタム ストラテジを作成し、SmartXplorer で実行する方法を示します。

演習

手順 1：演習プロジェクトを開く

lab2 ディレクトリにある stopwatch.xise プロジェクトを開きます。

手順 2：2 つの最良のストラテジの特定

演習 1 では、あらかじめ定義されたストラテジに対して次に示す結果が得られており、MapGlobOptLogOptRegDup と MapRegDup で最良の結果が得られたことがわかります。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffortIOReg	local-pc	run5	Done	6785	00h 00m 39s	Copy Results
MapGlobOptLogOptRegDup	local-pc	run2	Done	248	00h 00m 44s	Copy Results
MapRegDup	local-pc	run4	Done	1081	00h 00m 38s	Copy Results
MapRunTime	local-pc	run1	Done	1806	00h 00m 40s	Copy Results
MapGlobOptIOReg	local-pc	run3	Done	7690	00h 00m 42s	Copy Results
MapLogOptRegDup	local-pc	run6	Done	1612	00h 00m 39s	Copy Results
MapExtraEffort2	local-pc	run7	Done	1817	00h 00m 39s	Copy Results

手順 3：カスタム ストラテジ ファイルの作成

MapGlobOptLogOptRegDup および MapRegDup に基づくカスタム ストラテジを作成する前に、これらのストラテジで使用されているオプションを確認する必要があります。

この情報は、lab1\smartxplorer_results ディレクトリにある smartxplorer.txt ファイルに保存されています。このファイルには、前回実行されたストラテジのオプションがすべて含まれます。lab2 ディレクトリの lab1_smartxplorer.txt ファイルに、このファイルの内容が含まれています。


```
-----
Strategy : MapGlobOptLogOptRegDup
-----
Run index    : run2
Map options  : -ol high -xe n -global_opt speed -logic_opt on -
register_duplication on -w
Par options  : -ol high -xe n
...
-----
Strategy : MapRegDup
-----
Run index    : run4
Map options  : -ol high -xe n -register_duplication on -w
Par options  : -ol high -xe n
...
```

この情報を使用して、ストラテジ ファイルを作成します。MapGlobOptLogOptRegDup ストラテジの名前を My_Strat_1 に、MapRegDup ストラテジの名前を My_Strat_2 に変更します。

1. lab2 ディレクトリにある my_strategy.txt ファイルを開きます。このファイルに My_Strat_1 と My_Strat_2 の定義が含まれています。

```
{
  "spartan6":
  {
    "MAP-Par options":
    (
      { "name": "My_Strat_1",
        "map": " -ol high -xe n -global_opt speed -logic_opt on register_duplication on -w ",
        "par": " -ol high -xe n" },
      { "name": "My_Strat_2",
        "map": " -ol high -xe n -register_duplication on -w",
        "par": " -ol high -xe n" },
    ),
  },
}
```

2. [Tools] → [SmartXplorer] → [Launch SmartXplorer] をクリックするか、またはツールバーの

[Launch SmartXplorer] ボタン  をクリックします。

3. [Other SmartXplorer command line options] フィールドで、-sf オプションを使用して my_strategy.txt ファイルを指定します。
4. 2 つのカスタム ストラテジのみが実行されることを確実にするため、[Maximum number of runs] を 2 に設定します。

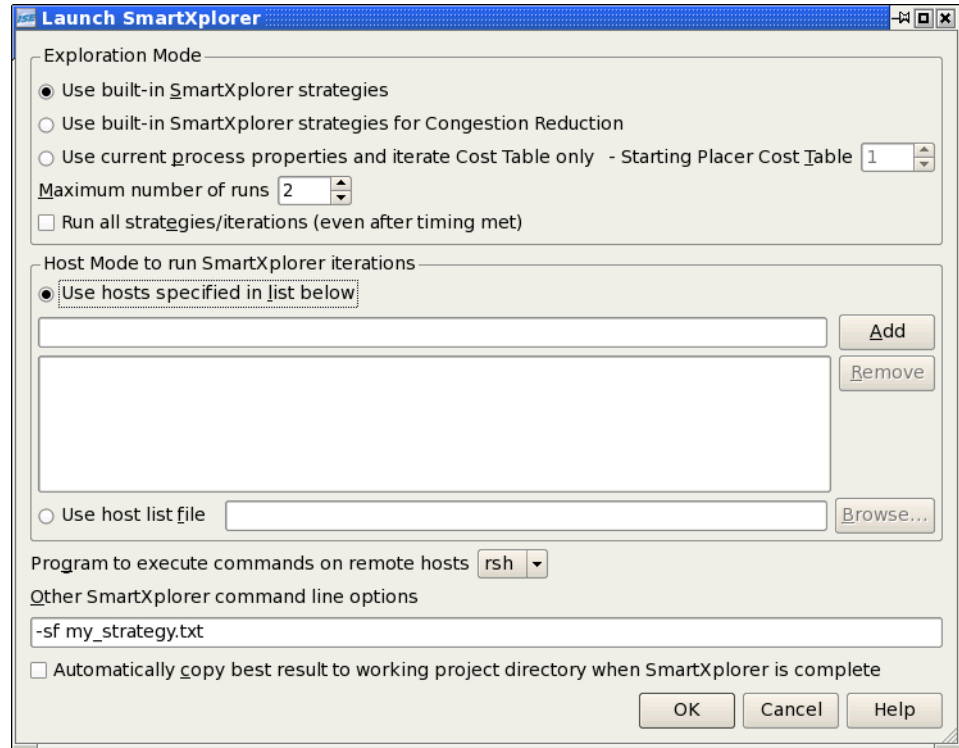


図 5-1 : SmartXplorer の [Maximum number of runs] と [Other SmartXplorer command line options] フィールド (Linux)

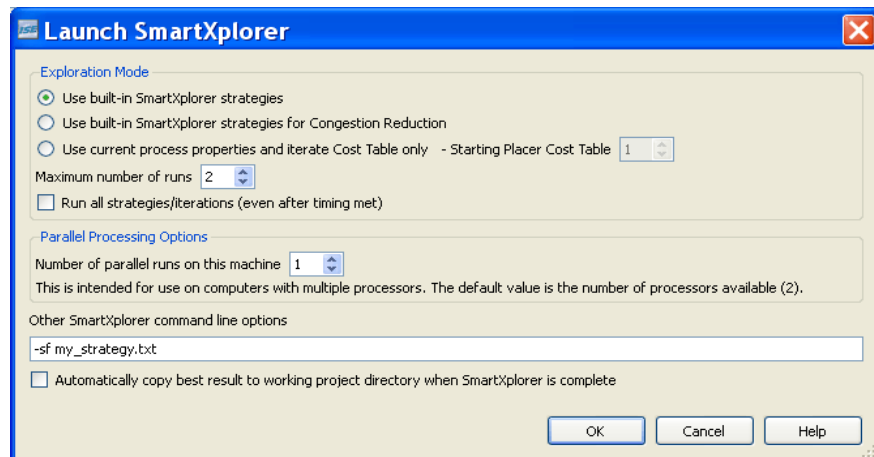


図 5-2 : SmartXplorer の [Maximum number of runs] と [Other SmartXplorer command line options] フィールド (Windows)

[Maximum number of runs] を 2 ではなく 5 に設定すると、あらかじめ定義されたストラテジでの場合と同様に、`my_strategy.txt` ファイルに含まれる 2 つのストラテジのうちよい方のストラテジが、異なるコスト テーブルを使用して追加で 3 回で実行されます。これは有効な方法です。

5. [OK] をクリックします。

次のような結果が表示されます。これは、演習 1 の結果と同じです。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
My_Strat_1	local-pc	run1	Done	248	00h 00m 44s	Copy Results
My_Strat_2	local-pc	run2	Done	1081	00h 00m 39s	Copy Results

メモ： カスタム ストラテジを使用する場合、[Copy Result] ボタンの操作は正しくサポートされていません。プロジェクト ディレクトリにファイルはコピーされますが、マップおよび配置配線のオプションはプロセス プロパティに正しくアップデートされません。手動で対応するオプションを設定する必要があります。

まとめ

この演習では、カスタム ストラテジ ファイルを作成し、それを使用してデザインを実行しました。

また、カスタム ストラテジのサポートとあらかじめ定義されているストラテジのサポートの違いも説明しました。

演習 3W：複数のストラテジの同時実行 (Windows)

重要： この演習は、マルチコア プロセッサまたは複数のプロセッサを持つ Windows マシン用です。

目標

SmartXplorer の主な機能の 1 つに、複数のストラテジを並列して同時に実行できるという機能があります。

この演習では、Windows マシンで SmartXplorer で同時に実行可能なストラテジの数を指定し、ランタイムがどれだけ短縮されるかを調べます。

演習

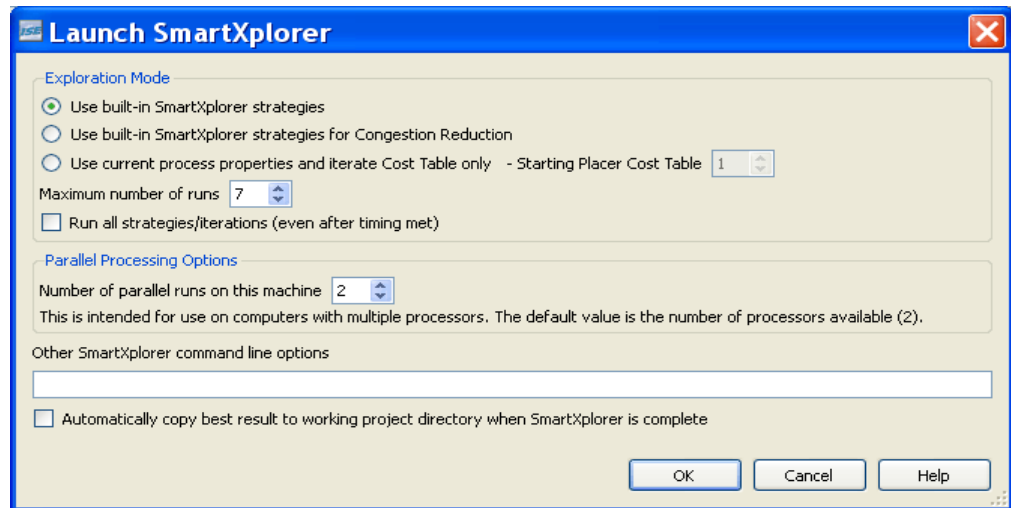
手順 1：演習プロジェクトを開く

lab3 ディレクトリにある stopwatch.xise プロジェクトを開きます。

手順 2：並列実行可能なストラテジ数を 2 に設定して SmartXplorer を実行

並列実行可能なストラテジの数を 2 に設定して、あらかじめ定義された 7 つのストラテジを実行します。

1. [Tools] → [SmartXplorer] → [Launch SmartXplorer] をクリックするか、またはツールバーの [Launch SmartXplorer] ボタンをクリックし、[Launch SmartXplorer] ダイアログ ボックスを開きます。
2. [Number of parallel runs on this machine] を 2 に設定します。



3. [OK] をクリックします。SmartXplorer で次の結果が生成されます。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffortIOReg	local-pc	run5	Done	6785	00h 00m 39s	Copy Results
MapGlobOptLogOptRegDup	local-pc	run2	Done	248	00h 00m 55s	Copy Results
MapRegDup	local-pc	run4	Done	1081	00h 00m 56s	Copy Results
MapRunTime	local-pc	run1	Done	1806	00h 00m 45s	Copy Results
MapGlobOptIOReg	local-pc	run3	Done	7690	00h 00m 45s	Copy Results
MapLogOptRegDup	local-pc	run6	Done	1612	00h 00m 50s	Copy Results
MapExtraEffort2	local-pc	run7	Done	1817	00h 00m 39s	Copy Results

4. この結果を、7 つのストラテジを 1 つずつ順番に実行した場合と比較すると、7 つのストラテジすべてを完了するのにかかる時間は 30% 短縮されています。
この結果は、次のコンフィギュレーションのマシンで得られたものです。

Intel(R) Core(TM)2 Duo CPU
T9500 @ 2.60GHz
2.59 GHz, 3.50 GB of RAM

メモ：1 つのストラテジが完了し、タイミングが満たされると、現在実行中のストラテジの実行がすべて停止します。これは、複数のストラテジが同時に実行されており、[Run all strategies/iterations] がオフの場合の動作です。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapGlobOptLogOptRegDupCT2	local-pc	run8	Done	492	00h 00m 46s	Copy Results
MapGlobOptLogOptRegDupCT3	local-pc	run9	Done	0	00h 00m 41s	Copy Results
MapGlobOptLogOptRegDupCT4	local-pc	run10	Stopped	0	00h 00m 33s	Copy Results
MapLogOptRegDup	local-pc	run6	Done	1612	00h 00m 50s	Copy Results

上の図で、グレーの長方形で囲まれている 2 つのストラテジのステータスを見てみます。

MapGlobOptLogOptRegDupCT3 の後に MapGlobOptLogOptRegDupCT4 が開始していますが、MapGlobOptLogOptRegDupCT4 が完了する前に MapGlobOptLogOptRegDupCT3 でタイミングが満たされたため、MapGlobOptLogOptRegDupCT4 の実行が停止され、[Status] 列に [Stopped] と示されています。

まとめ

この演習では、Microsoft Windows マシンで複数のストラテジを同時に実行する方法を学びました。

SmartXplorer で 2 つのストラテジを同時に実行することにより、全体的なランタイムが 30% 短縮しました。

メモ：実際の短縮時間は、マシンのコンフィギュレーションによって異なります。

演習 3L：複数のストラテジの同時実行 (Linux)

重要： この演習は、Linux ネットワーク、LSF および SGE コンピュータ ファーム、マルチコア プロセッサまたは複数のプロセッサを持つマシン用です。

目標

この演習では、通常の Linux ネットワーク、LSF および SGE コンピュータ ファーム、マルチコア プロセッサまたは複数のプロセッサを持つマシンでザイリンクス環境を設定し、SmartXplorer を使用する方法を説明します。

これらのネットワークで SmartXplorer を使用する場合は考慮事項を示します。

演習

手順 1：ザイリンクス環境と結果の保存

SmartXplorer を起動する前に、次の 2 つの事項を理解する必要があります。

- 各マシンでのザイリンクス ソフトウェアの環境設定
- SmartXplorer の結果の保存場所

1. ザイリンクス環境の設定

デザイン ストラテジの実行に 3 つの マシン (L1、L2、および L3) を使用する通常の Linux ネットワークの場合を考えてみます。ISE Project Navigator および SmartXplorer は、L1 から起動します。

L2 (L3) でジョブを起動する前に、SmartXplorer により自動的に L2 (L3) マシンで \$XILINX 環境変数が L1 の \$XILINX と同じ値に設定されます。これは、次のことを意味します。

- ◆ ザイリンクス ソフトウェアがネットワーク上にインストールされている場合、L2 (L3) でそのソフトウェアにアクセスする必要があり、また L1 に定義されたネットワーク パスがすべてのマシンで有効となるよう同じネットワーク割り当てポイントを使用する必要があります。
- ◆ ザイリンクス ソフトウェアが L1 のローカル ディスクにインストールされている場合、L2 (L3) のローカル ディスクの L1 と同じパスに同じバージョンのザイリンクス ソフトウェアがインストールされていることが必要です。

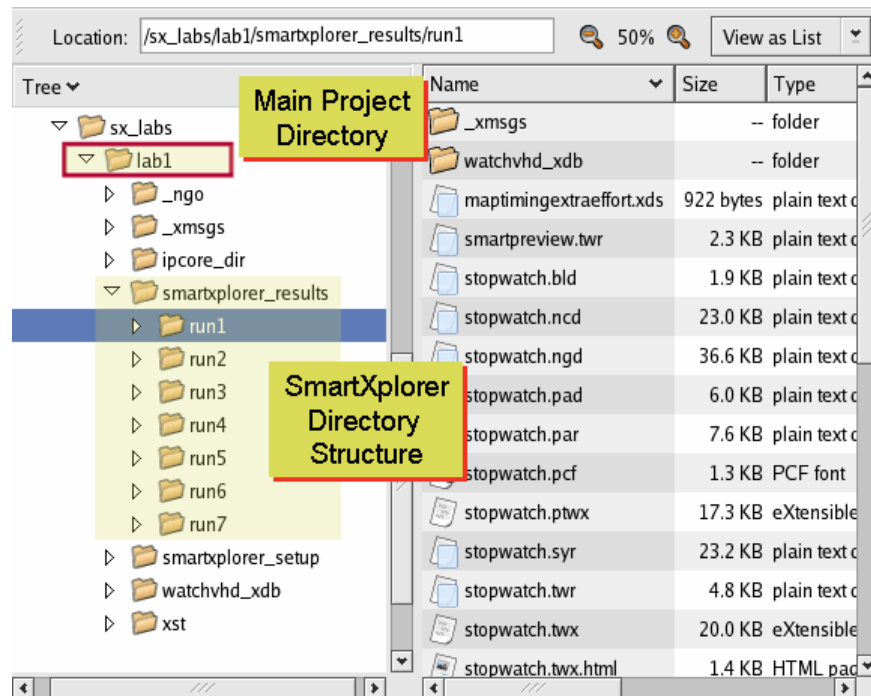
環境変数がこのように設定されるのは、各デザイン ストラテジが同じ条件下で実行されるようにするためです。\$XILINX に加え、L1 で設定されているすべてのザイリンクス環境変数 (接頭辞「XIL_」で始まるもの) が検出され、L2 および L3 に適用されます。

LSF および SGE コンピュータ ファームを使用する場合でも、ザイリンクス ソフトウェアを実行する各マシンで、ISE Project Navigator を実行したマシンと同じザイリンクス環境変数が設定されている必要があります。

2. 結果の保存場所

演習 1 で説明したように、SmartXplorer の結果はすべて smartxplorer_results という別ディレクトリに保存されます。

複数のマシンを使用する場合でも同じです。デフォルトでは、SmartXplorer の結果はすべて ISE Project Navigator を実行しているマシンに保存されます。そのため、すべてのマシンにこのディスク エリアに対する読み取り/書き込み権限が必要です。-wd オプションを使用して、異なる保存場所を指定することも可能です。



手順 2：実行前のチェックリスト

1. 使用するマシンのリストを作成します。
2. 各マシンで同じザイリンクス環境が設定されていることを確認します。
3. すべてのマシンに SmartXplorer の結果が保存されるディレクトリへの読み出し/書き込み権限があることを確認します。

手順 3：演習プロジェクトを開く

lab3 ディレクトリにある stopwatch.xise プロジェクトを開きます。

手順 4：通常の Linux ネットワークでの SmartXplorer の実行


2 つの Linux マシン xgr-sweng109 および xgr-sweng135 を使用して、あらかじめ定義された 7 つのストラテジを実行します。xgr-sweng109 上では 1 つのジョブ、xgr-sweng135 では 2 つのジョブを同時に実行できるとします。ISE Project Navigator の実行には、xgr-sweng135 を使用します。

1. lab3 ディレクトリにある my_strategy.txt ファイルをテキスト エディタで開きます。このファイルには、デザイン ストラテジを実行するのに使用するマシンがリストされています。このファイルには、各マシンの名前を個別の行に記述します。

```
xgr-sweng135
xgr-sweng135
xgr-sweng109
```

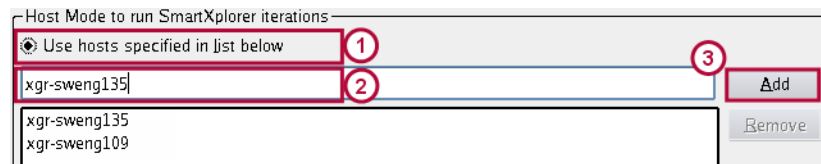
同じマシンで 2 つのジョブを同時に実行できるようにするには、上記の例の xgr-sweng135 のように、そのマシンを 2 回リストする必要があります。

メモ： my_hostlist.txt ファイルで xgr-sweng135 および xgr-sweng109 を実際に使用するマシンの名前に置き換え、ファイルを保存します。

2. [Tools] → [SmartXplorer] → [Launch SmartXplorer] をクリックするか、またはツールバーの [Launch SmartXplorer] ボタン  をクリックします。

3. [Use host list file] で my_hostlist.txt ファイルを指定します。

メモ： [Launch SmartXplorer] ダイアログ ボックスで [Use hosts specified in list below] をオンにして、マシン名を直接指定することも可能です。



4. [OK] をクリックします。次のような表が表示されます。xgr-sweng135 で 7 つのうち 4 つのストラテジが実行されました。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapExtraEffortIOReg	xgr-sweng135	run5	Done	6785	00h 02m 17s	Copy Results
MapGlobOptLogOptRegDup	xgr-sweng135	run2	Done	248	00h 03m 11s	Copy Results
MapRegDup	xgr-sweng109	run4	Done	1081	00h 01m 37s	Copy Results
MapRunTime	xgr-sweng135	run1	Done	1806	00h 02m 46s	Copy Results
MapGlobOptIOReg	xgr-sweng109	run3	Done	5256	00h 02m 14s	Copy Results
MapLogOptRegDup	xgr-sweng135	run6	Done	1612	00h 02m 07s	Copy Results
MapExtraEffort2	xgr-sweng109	run7	Done	1817	00h 01m 27s	Copy Results

メモ： 1 つのストラテジが完了し、タイミングが満たされると、現在実行中のストラテジの実行がすべて停止します。これは、複数のストラテジが同時に実行されており、[Run all strategies/iterations] がオフの場合の動作です。

Strategy	Host	Output	Status	Timing Score	Run time	Copy Results
MapGlobOptLogOptRegDupCT2	xgr-sweng135	run8	Done	492	00h 02m 39s	Copy Results
MapGlobOptLogOptRegDupCT3	xgr-sweng109	run9	Done	0	00h 02m 11s	Copy Results
MapGlobOptLogOptRegDupCT4	xgr-sweng135	run10	Stopped	0	00h 00m 41s	Copy Results
MapLogOptRegDup	xgr-sweng135	run6	Done	1612	00h 01m 21s	Copy Results

上の図で、グレーの長方形で囲まれている 2 つのストラテジのステータスを見てみます。
 MapGlobOptLogOptRegDupCT3 の後に MapGlobOptLogOptRegDupCT4 が開始していますが、
 MapGlobOptLogOptRegDupCT4 が完了する前に MapGlobOptLogOptRegDupCT3 でタイミング
 が満たされたため、MapGlobOptLogOptRegDupCT4 の実行が停止され、[Status] 列に [Stopped] と
 示されています。

手順 5：LSF または SGE での SmartXplorer の実行

LSF および SGE コンピュータ ファームを使用するのは、通常の Linux ネットワークを使用するの
 とほとんど同じで、ホスト リスト ファイルを作成して [Use host list file] で指定します。

主な違いは、ホスト リスト ファイルでのホスト マシンの定義方法です。LSF および SGE コン
 ピュータ ファームのフォーマットは、次のとおりです。

LSF	:LSF {"queue_name": "MYQUEUE", "max_concurrent_runs":N, "bsub_options": "additional_options" }
SGE	:SGE {"queue_name": "MYQUEUE", "max_concurrent_runs":N, "qsub_options": "additional_options" }

説明：

- queue_name：キューの名前を指定します。「MYQUEUE」を LSF または SGE キューの名前に置き換えてください。
- max_concurrent_runs：同時に実行可能なジョブの最大数を指定します。N に正の整数を指定してください。
- bsub_options：追加の LSF オプションを指定します。「additional_options」を LSF オプションに置き換えてください。オプションを使用しない場合は、“” を使用します。
- qsub_options：追加の SGE オプションを指定します。「additional_options」を SGE オプションに置き換えてください。オプションを使用しない場合は、“” を使用します。

例：

キューの名前が lin64_q、同時に実行可能なジョブの最大数が 6、指定する LSF または SGE オプションがない場合は、ホスト リスト ファイルに次のように記述します。

LSF	:LSF {"queue_name": "lin64_q", "max_concurrent_runs":6, "bsub_options": "" }
SGE	:SGE {"queue_name": "lin64_q", "max_concurrent_runs":6, "qsub_options": "" }

まとめ

この演習では、複数の Linux マシンで SmartXplorer を使用して複数のデザイン ストラテジを同時に実行する方法を説明しました。

これらのネットワークで SmartXplorer を使用する場合の考慮事項を示しました。

カスタム ファイル

目標

この付録では、次のものの例を示します。

- [カスタム ストラテジ ファイル](#)
- [ホスト リスト ファイル \(Linux\)](#)
- [さまざまなタスクでの SmartXplorer の設定](#)

カスタム ストラテジ ファイル

次に、カスタム ストラテジ ファイルの例を示します。Spartan®-6 デバイスの 2 つのストラテジと Virtex®-5 デバイスの 2 つのストラテジが含まれています。

```
{
  "spartan6":
  {
    "MAP-Par options":
    (
      { "name": "My_Strat_1",
        "map": " -ol high -xe n -global_opt speed -logic_opt on -
register_duplication on -w ",
        "par": " -ol high -xe n" },
      { "name": "My_Strat_2",
        "map": " -ol high -xe n -register_duplication on -w ",
        "par": " -ol high -xe n" },
    ),
  },
  "virtex5":
  {
    "MAP-Par options":
    (
      { "name": "My_Strat_3",
        "map": " -ol high -xe n -pr b -t 7 -w ",
        "par": " -ol high -xe n" },
      { "name": "My_Strat_4",
        "map": " -ol high -xe n -t 3 -w ",
        "par": " -ol high -xe n" },
    ),
  },
}
```

ホスト リスト ファイル (Linux)

通常の Linux ネットワーク

次に、通常の Linux ネットワーク用のホスト リスト ファイルの例を示します。

```
lin_machine_1  
lin_machine_1  
lin_machine_2  
lin_machine_3
```

この例では、SmartXplorer で 4 つのストラテジが同時に実行されます。3 つの Linux マシンが使用され、lin_machine_1 では 2 つのストラテジが同時に実行されます。

LSF コンピュータ ファーム

次に、LSF コンピュータ ファーム用のホスト リスト ファイルの例を示します。この例では、キュー名は lin64_q で、同時に実行可能なジョブの最大数は 6 です。

```
:LSF {"queue_name":"lin64_q", "max_concurrent_runs":6, "bsub_options": ""}
```

SGE コンピュータ ファーム

次に、SGE コンピュータ ファーム用のホスト リスト ファイルの例を示します。この例では、キュー名は lin64_q で、同時に実行可能なジョブの最大数は 6 です。

```
:SGE {"queue_name":"lin64_q", "max_concurrent_runs":6, "qsub_options": ""}
```


さまざまなタスクでの SmartXplorer の設定

タスク 1：あらかじめ定義されているストラテジすべてを実行

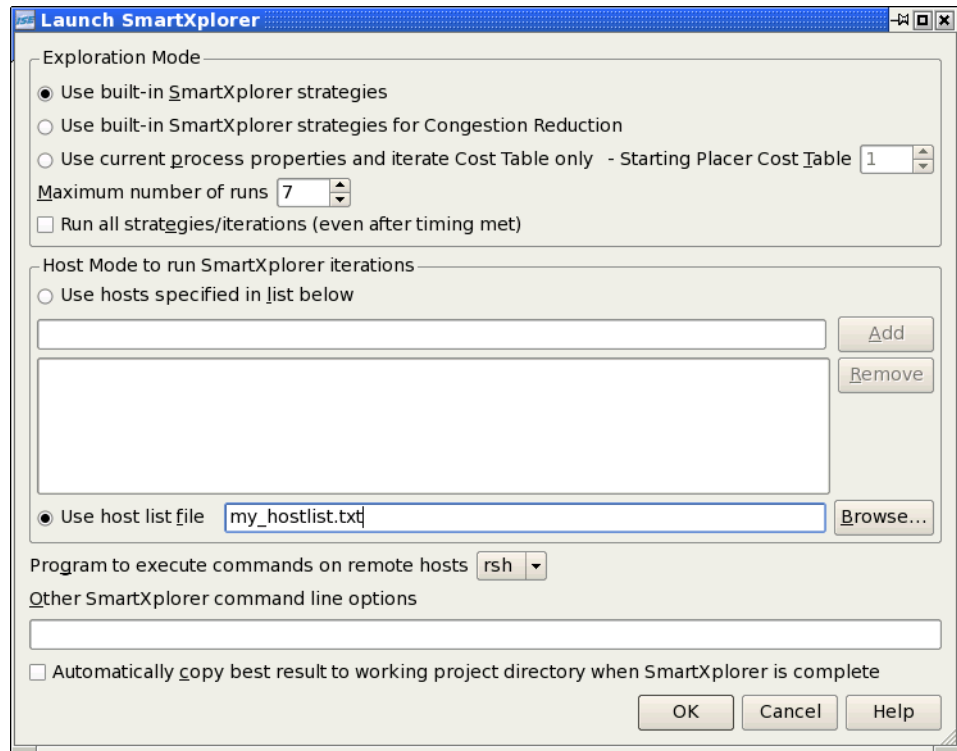


図 A-1：あらかじめ定義されているストラテジすべてを実行 (Linux)

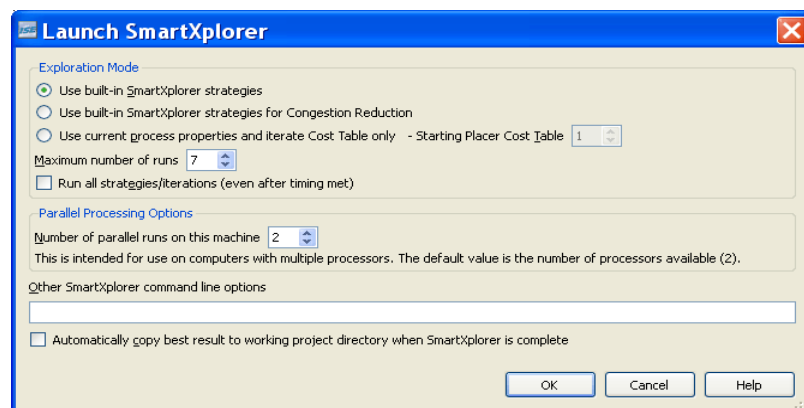


図 A-2：あらかじめ定義されているストラテジすべてを実行 (Windows)

この例は、あらかじめ定義されている 7 つのストラテジすべてを実行します ([Maximum number of runs] を 7 に設定)。Linux の例では、ホスト リスト ファイルで指定されているマシン上でストラテジが実行されます。Windows の例では、同じホスト上で 2 つのストラテジが同時に実行されます。タイミングが満たされると、実行は終了します。

タスク 2：あらかじめ定義されているストラテジの最初の 3 つを実行

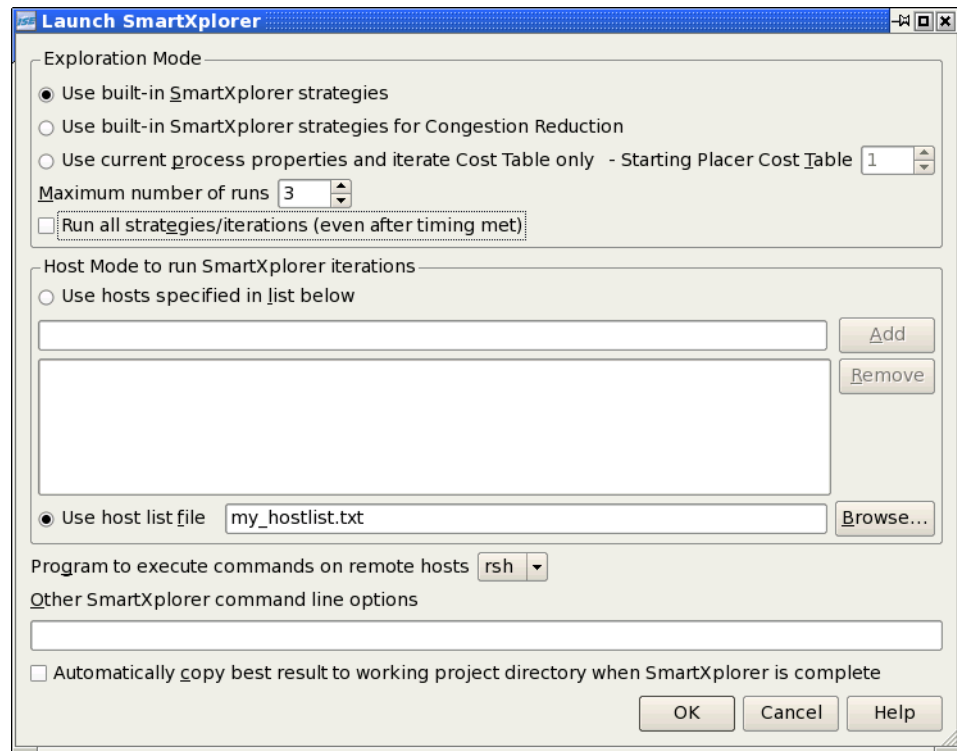


図 A-3：あらかじめ定義されているストラテジの最初の 3 つを実行 (Linux)

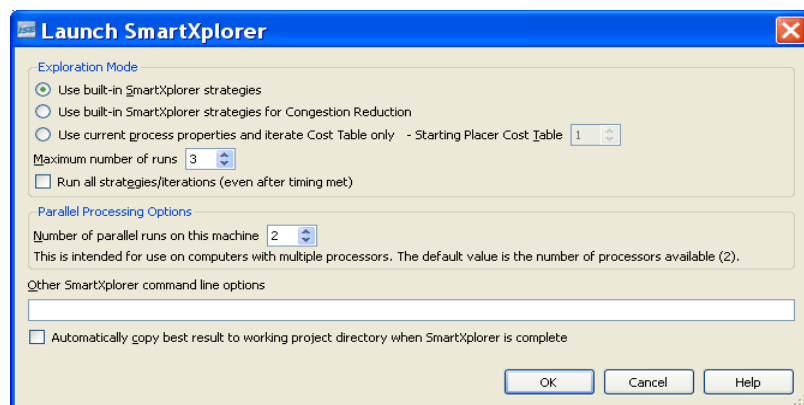


図 A-4：あらかじめ定義されているストラテジの最初の 3 つを実行 (Windows)

この例は、あらかじめ定義されているストラテジのうち最初の 3 つのみを実行します ([Maximum number of runs] を 3 に設定)。Linux の例では、ホスト リスト ファイルで指定されているマシン上でストラテジが実行されます。Windows の例では、同じホスト上で 2 つのストラテジが同時に実行されます。タイミングが満たされると、実行は終了します。

タスク 3： あらかじめ定義されているストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 5 回実行

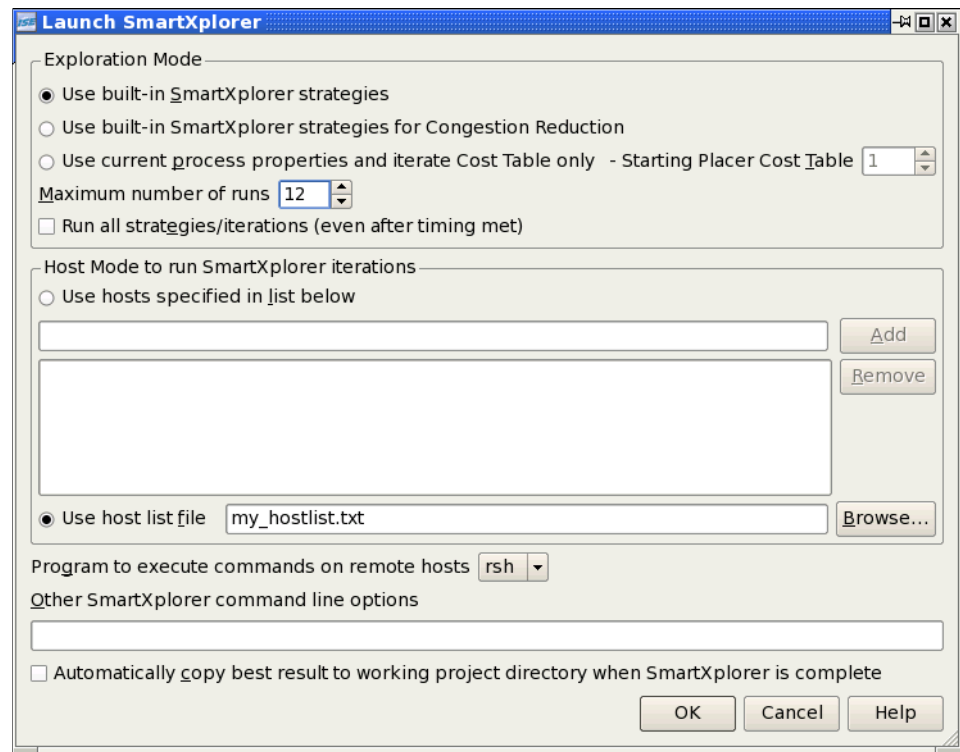


図 A-5： あらかじめ定義されているストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 5 回実行 (Linux)

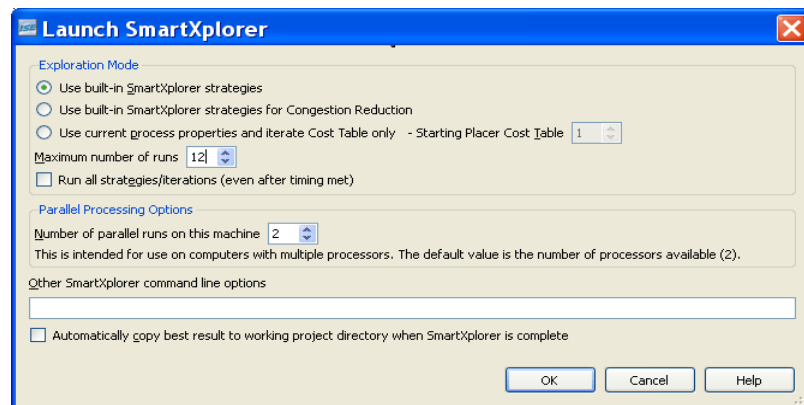


図 A-6： あらかじめ定義されているストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 5 回実行 (Windows)

段階 1： あらかじめ定義されている 7 つのストラテジすべてを実行します。

段階 2： 段階 1 から最良のストラテジを選択し、そのストラテジを異なるコスト テーブルを使用して追加で 5 回実行します ([Maximum number of runs] を $7 + 5 = 12$ に設定)。

Linux の例では、ホスト リスト ファイルで指定されているマシン上でストラテジが実行されます。
Windows の例では、同じホスト上で 2 つのストラテジが同時に実行されます。

タイミングが満たされると、実行は終了します。

タスク 4：カスタム ストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 3 回実行

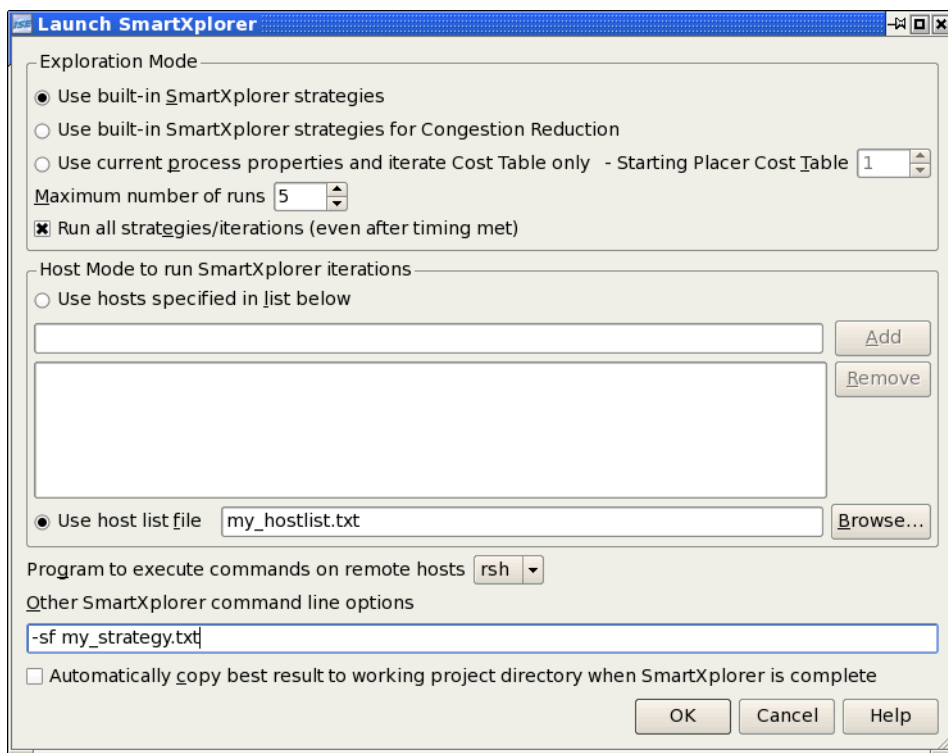


図 A-7：カスタム ストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 3 回実行 (Linux)

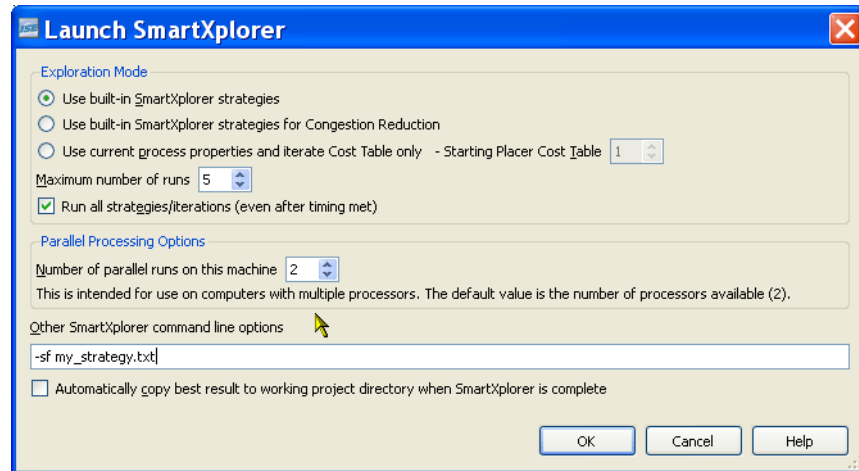


図 A-8 : カスタム ストラテジすべてを実行し、異なるコスト テーブルを使用して追加で 3 回実行 (Windows)

この例では、カスタム ストラテジ ファイルに 2 つのストラテジのみが含まれているとします。

段階 1 : カスタム ストラテジ ファイルで指定されている 2 つのストラテジを実行します ([Other SmartXplorer command line options] で `-sf` オプションを指定)。

段階 2 : 段階 1 から最良のストラテジを選択し、そのストラテジを異なるコスト テーブルを使用して追加で 3 回実行します ([Maximum number of runs] を $2 + 3 = 5$ に設定)。

Linux の例では、ホスト リスト ファイルで指定されているマシン上でストラテジが実行されます。Windows の例では、同じホスト上で 2 つのストラテジが同時に実行されます。

タイミングが満たされた場合でも、5 つのストラテジがすべて実行されます ([Run all strategies/iterations] をオンに設定)。

