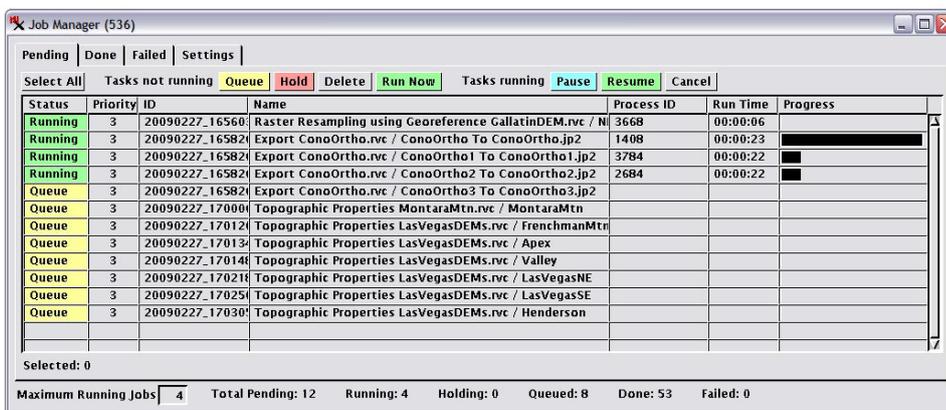


TNTmips のジョブ処理システム

TNTmips のジョブ処理システム (JPS) および関連のジョブマネージャは、ユーザの生産プロジェクトのスループット (処理量) を飛躍的に高め、コンピュータのマルチコアを利用し、他のソフトウェア製品やインターネットと相互利用するように設計されています。タスクは TNTmips の対話処理や地理空間スクリプト (SML)、ウェブアプリケーション、ユーザ独自のプログラムによって定義することができます。各タスクは XML のジョブファイルであり、入力と出力パラメータや TNT 処理、SML スクリプト、またはジョブファイルがコントロール・実行する他のプログラムを定義します。TNTmips の実行中、TNT ジョブ処理システム (JPS) はバックグラウンドで自動的に活性化され、これらの TNT タスクを見つけて優先順位をつけ、並行処理します。タスクが実行中でないときは、このバックグラウンド処理はコンピュータリソースをほとんど使いません。

作成されたそれぞれのタスクの XML ジョブファイルは、ジョブディレクトリの中に書き込まれます。TNTmips のジョブマネージャはこのディレクトリの中の全ての現在実行中のタスクを管理するインタフェースを提供します。ジョブマネージャを使うと各ジョブに優先度を付けたり、いくつかのタスクを並行処理するか設定したり、いつタスクを実行するかをスケジュールしたり、各タスクのステータスを確認したりできます (詳しくはテクニカルガイドの“システム: ジョブ処理の管理 (System: Managing Job Processing)”をご覧ください)。



TNTmips のジョブマネージャは新しいジョブファイルが作られると (まだ開いていなければ) 自動的に開きます。ジョブマネージャを使うと処理待ちのジョブを選びそのステータスを変更したり (停止 (hold)、再開 (release)、削除、次の処理スレッドでの実行にセット)、処理時間や現在実行中のジョブの進捗バーを表示したりできます。完了および失敗したジョブの一覧は別のタブパネルに表示されます。ジョブ処理の時刻・曜日のスケジューリングをセットできます。

JPS (ジョブ処理システム) では、多くの同じまたはミックスした TNT 処理をバックグラウンドで並行して実行できます。JPS タスクがバックグラウンドで実行中であっても、他の TNT プロセスやアプリケーションの操作を続けることが可能です。マルチコアのコンピュータを使えば、同時に実行するタスクの数を定義できます。シングルコアのマシンであっても、JPS はいくつかのタスクを並行処理することでスループットを上げることができます。特に、タスクをコンピュータをあまり使用していない時にスタートしたり、TNTmips がインストールされている他のコンピュータを使用したりするようにプログラムすれば効果的です。

対話式の TNT 処理の利用

ジョブ処理をサポートするために、いくつかの TNT 処理が修正され、新しい実行ボタンが追加されました。画面には、(JPS をバイパスするための) 標準の

Run... Run Job... Save Job...

[実行 (Run)] ボタンに加えて、[ジョブの実行 (Run Job)] ボタンと [ジョブの保存 (Save Job)] ボタンが増えました。新しい [ジョブの実行] ボタンや [ジョブの保存] ボタンを選択すると、TNT 処理で定義したタスクや複数タスクがジョブディレクトリの中に1つまたは複数個の XML ジョブファイルを生成します。ジョブを実行すると、それらは並行処理を待つタスクキューの一番上に加えられます。ジョブを保存すると、それらは停止 (hold) ステータスのタスクキューに加えられます。これらのボタンのどちらかを選ぶと、(まだ開いていなければ) ジョブマネージャ画面が自動的に開き、処理待ちまたは進行中の他のタスクに影響が出ないよう新規タスクを調節することができます。

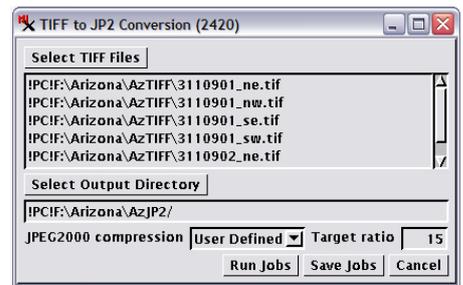
バッチ処理をサポートする TNT 処理はその処理が実行される都度ジョブファイルをセットアップします。インポートおよびエクスポート処理は各インポートまたはエクスポートの操作の都度、自動的に別のジョブファイルを生成します。例えば、1000 個の MrSID ファイルを JP2 ファイルにエクスポートするとしたら、ジョブキューの中にはに停止ステータスまたはキュー (queue) ステータスを持つ 1000 個の XML ジョブファイルができます。ジョブ処理の簡単な使用例の説明が、テクニカルガイドの“エクスポート: ジョブ処理を使用した同時エクスポート (Export: Concurrent Exports Using Job Processing)”にあります。

ペアのスクリプトの使用

(入出力に関わる I/O スクリプトとプロセスに関わる P スクリプトの) 一組の SML スクリプトを使って、多数のジョブファイルをセットアップし、個別の生産タスクを達成することができます。I/O スクリプトは、対話的ユーザインタフェースを提供し、1つまたは多数のジョブファイルをセットアップします。

ある SML クラスがあって、スクリプトから XML ジョブファイルを書き出すことができます。I/O スクリプトによって

作られる各ジョブファイルには、タスク用の入出力パラメータと実行する P スクリプトの識別情報が含まれます。典型的な応用例として、3 つ以上のバンドのマルチバンドの GeoTIFF イメージ 1000 個を同等のマルチバンド GeoJP2 イメージに変換する P スクリプトがあります。I/O スクリプトでは、入力の GeoTIFF ファイルの一括選択と出力の GeoJP2 ファイルのネーミング（または自動ネーミング）やファイルの格納場所を指定します。この用途のサンプル I/O スクリプトと P スクリプトが、テクニカルガイドの“システム：地理空間スクリプトを使ったカスタムジョブ処理 (System: Custom Job Processing with Geospatial Scripts)”に記載されています。

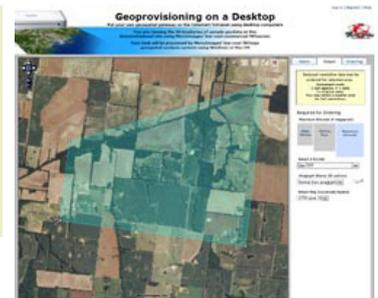
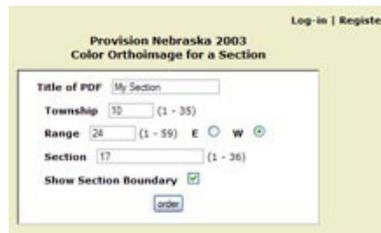


外部ソースからのジョブ生成

各ジョブファイルには JPS が実行する TNT 処理や SML スクリプトの識別情報と使われる入出力パラメータとが含まれます。XML ジョブファイルの中の入出力パラメータは、ウェブアプリケーションや、BASIC や JAVA 等の任意の言語で書かれたカスタムプログラムで生成可能です。そして、これらのアプリケーションではセンサーやその他のデータ収集システムのような個別のソースも参照できます。ジョブファイルを作るこれらのどの方法でも、SML スクリプトか TNT プロセスにより定義されたタスクは即座に実行されるか、待ち行列に入るか、スケジュールに組み込まれます。あなたの地理空間アプリケーションはたいへん複雑なものかもしれません。例えば、それはセンサーネットワークや車列をモニターし、逐一表示するものかもしれません。他のジョブファイルを作るかもしれません。ユーザの指示と要求に基づいたカスタムレポートと地図を提供するものかもしれません。XML ジョブファイルの適切なフォーマットはテクニカルガイドの“システム：ジョブファイルとは？ (System: What Are Job Files?)”に記載されています。

地理データ処理の提供

マイクロイメージ社の見本の地理データ処理提供サイトでは、ジョブ処理を使って各ユーザのリクエストを実行しています。ここで挙げた様々なウェブクライアントは無料の MySQL や PHP を使用してジョブファイルを生成しています。これらのジョブファイルは適切な SML スクリプトを呼び出し、HTML フォームやウェブアプリケーションより提供される入力データをもとにジオプロビジョニング（地理情報処理の提供）タスクを実行します。この処理スクリプトは、E メールメッセージを送ってユーザにその結果を通知することもできます。データ処理提供サイト www.geoprovisioning.com をお試しください。



上に挙げたようなビジュアル（画像表示が可能な）ジオプロビジョニングクライアントは、TNT タイルセットの参照画像を表示し、グラフィカルツールを使ってズームやパン、抜き出しのための矩形またはポリゴン領域を選択することができます。TNT タイルセットは TNTserver がアクセスするファイルサーバシステム上に置かれ、ビジュアルクライアントに対して参照画像を提供します。クライアントにより開始された SML スクリプトはこれらの同じタイルセットから直接データを抜き出します。ジオプロビジョニング用に TNTserver を必要としないビジュアルクライアントでは Google マップをベースとしたウェブアプリケーションとしてセットアップすることも可能です。このウェブアプリケーションは参照画像として、TNT タイルセットがインターネットを使って処理された地理空間データから、TNTmips のモザイク処理を使って作成したバラバラの Google マップタイルオーバーレイを表示したり、領域を選択するために必要なグラフィカルツールを提供します。JPEG 圧縮の使用のため Google マップのタイルオーバーレイでは画質が低下するかもしれませんが、Google マップの構造は高速表示用に最適化されており、Google マップのインタフェースはより使い易く改造されています。そのため Google マップのタイルオーバーレイにアクセスするための (TNTserver のような) 特別な地理データ管理サーバは必要としません。Google マップベースのウェブアプリケーションはエリア選択と他の処理パラメータを収集し、XML ジョブファイルに書き出して SML スクリプトを呼び出し、オリジナルの TNT オブジェクトやファイルサーバ上のタイルセットにマッチさせた上で要求された処理を実行します。

他のアプリケーションの起動

SML スクリプトは簡単な関数を呼び出すことにより他のアプリケーションプログラムを起動し、それにコマンド行のパラメータを引き渡すことができます。このように、ジョブ処理により実行したスクリプトは地理空間データやその他のデータを生成し、それらを他のアプリケーションに渡すことができます。例えば、あるジョブファイルから KML ファイルを生成する SML スクリプトを実行し、それを使って Google Earth を起動することもできます。また、あるスクリプトでは、表形式のデータを CSV ファイルに書き出し、それでスプレッドシートプログラムを起動したり、あるいはローカルのカスタムプログラムやウェブアプリケーションに渡して処理したり配布したりできます。このように、TNTmips のユーザは、TNTmips のジョブ処理や SML、ローカルまたはウェブ上のカスタムアプリケーションを連結して、複雑な地理空間解析を実行し、スプレッドシートやデータベースなどの他の商用プログラムへ情報を転送し、PDF や SVG などの一般的なフォーマットで解析結果を生成することができます。