

## VBAプログラミングによる自動画像挿入マクロの設計

石井, 大輔  
九州大学応用力学研究所技術室

<https://hdl.handle.net/2324/17078>

---

出版情報 : 九州大学応用力学研究所技術職員技術レポート. 11, pp.1-8, 2010-03. Research Institute for Applied Mechanics, Kyushu University

バージョン :

権利関係 :

# VBA プログラミングによる自動画像挿入マクロの設計

九州大学応用力学研究所技術室 石井 大輔

## 1. 緒言

近年、学会や技術研究会等での講演時には必ずと言ってよい程、プロジェクトを介した電子ファイルによる発表・操作が行なわれる。そのプレゼンテーションによく利用するソフトウェアの一つとして、Microsoft PowerPoint を挙げる人は多いであろう（他には、OpenOffice.org Impress (<http://ja.openoffice.org/>) や Adobe Acrobat/Acrobat Reader (<http://www.adobe.com/jp/products/reader/>) など）。

例えば、当該ソフトのスライドシート上に複数の写真や静止画像、研究結果等を順序よく並べて表示させたい場合、それら写真や図のファイルを一度個別に起動させ、写真や図のコピー/貼り付け作業を実施するか、もしくは PowerPoint の図挿入機能を利用して写真等を選択・貼付し、トリミングやサイズ縮尺を施しながら所望の位置に配置するのが一般的な作業であろう。少数画像なら手動作業もさほど手間ではないかもしれないが、何十枚も複数のシートに及ぶ場合にはその単純作業にかなりの時間と労力が割かれてしまう。特に、写真や図のトリミング・縮尺作業はその典型で、意外に作業時間を費やしてしまうことが多い。

筆者は、海洋観測データを解析した結果を1枚の PowerPoint スライドシートに貼り付けて発表用資料や打ち合わせ資料を作成する機会が多く、今までは1画像に対して画像コピー・シートへの画像貼り付け・トリミング・サイズ調整・所望位置への移動、といった一連の作業を毎回淡々と進めていた。しかし、当該作業は非常に効率が悪いので、この一連の作業を自動化する方法があるのではないかと模索し始めたのが本開発の端緒である。

本稿は、VBA (Visual Basic for Applications) プログラミングによって PowerPoint スライドシート上に自動で画像挿入可能なマクロの設計・開発を目的とし、以下に本機能を PowerPoint で活用した事例や実際の設計フォーム等について説明する。

## 2. VBA マクロ開発環境

本マクロは、Microsoft 社製 Office 2007 に包含されるプレゼンテーションソフト PowerPoint 2007 を使用し設計・作製した。マクロとは、ソフトウェア上で特定の操作手順をプログラムとして記述し自動化する機能のことである。そして、そのプログラムの記述に使用する言語はマクロ言語と呼ばれ、VB (Visual Basic) を基に改良されたものが VBA である。1994年に Excel 5.0 とともに登場した VBA は、様々な仕様拡張を経て、現在では Word や Excel 等を包含する Office 製品に共通したプログラミング言語として確立し、シリアルポート（例えば、RS-232C）を介した計測機器制御等にも定評がある。

VBA プログラミングを行なう環境は、Fig. 1 に示すような PowerPoint に付属の VB エディタ (VBE: Visual Basic Editor) である。そして、選択リストや実行/終了ボタン等のコントロールと呼ばれる視覚的部品を、基本となるフォームに配置し、所望の要求を満たすようにコ

ントロールの動作や書式等（メソッドやプロパティ等）を記述することで1つのプロジェクトを構成するのが、一連のVBA設計フローである。筆者は、今までにMicrosoft ExcelでVBAによるマクロ開発を行なった経験があるので、基礎知識をある程度備えた状態で本設計に臨むことができた。なお、既述したVBA専門用語やVBAとOffice製品との連携/諸動作、既存のVBAマクロ開発事例の詳細については、石井（2006, 2009）および参考文献に記載の専門書等を参照されたい。

Office 2007は、保存するファイルの種類として新しく採用された拡張子（docx, xlsx, pptx等）の他に、マクロを含むか否かを判別できるマクロ有効拡張子（docm, xlsx, pptm等）が導入された。今回、VBAプログラミングによるマクロ作製ということから、保存するPowerPointのファイル種類は「PowerPointマクロ有効プレゼンテーション（\*.pptm）」とした。

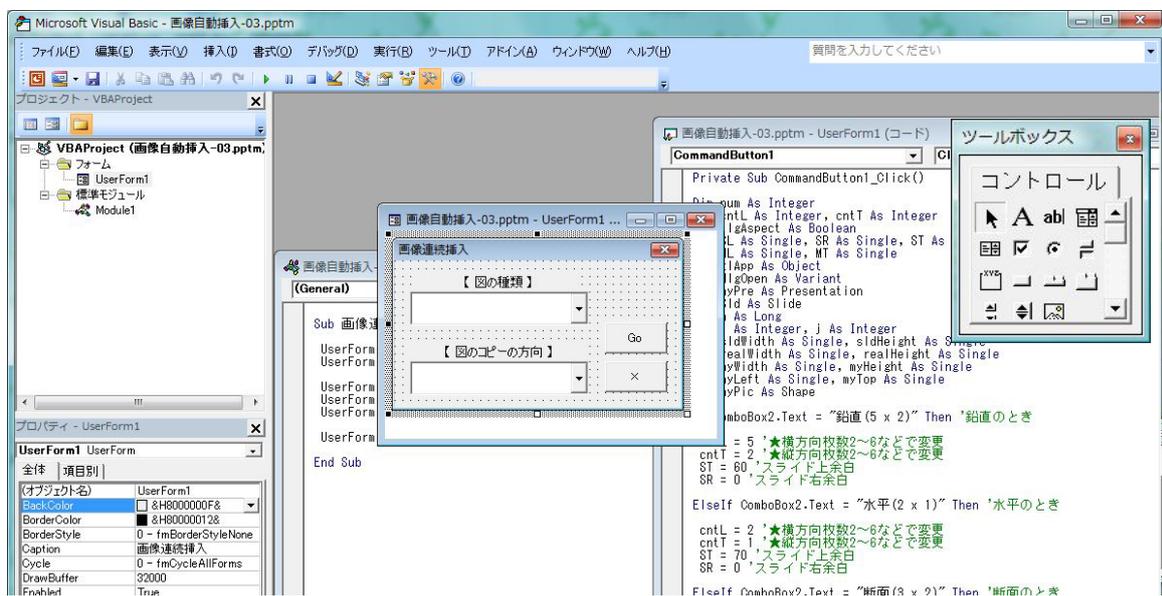


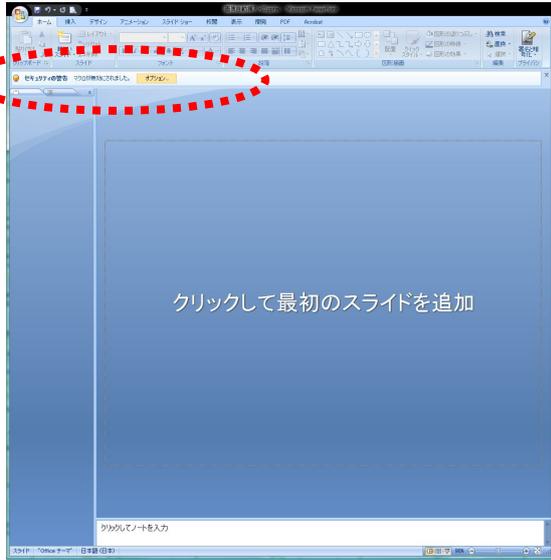
Fig. 1 VBAプログラミング環境をもつVBエディタの外観

### 3. 自動画像挿入マクロの基本動作

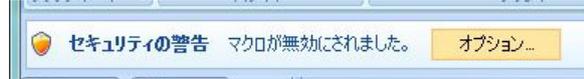
以下に、Microsoft PowerPoint VBAを利用した自動画像挿入マクロの基本動作について順に説明する。なお本稿は、PowerPoint 2007を使用して話しを進めるが、下位バージョン（ver. 2003以前）においても本質的には大きくは変わらず、互換性については特に問題ない。

まず、本稿で開発した自動画像挿入マクロを含むPowerPoint 2007を起動すると、Fig. 2 (a)のような初期状態となる。ユーザによるマクロの有効/無効の初期設定にもよるが、通常は点線で囲まれた上部に、Fig. 2 (b)に示すマクロセキュリティの警告表示が表示される。そのため、「信頼できるマクロ」を実行する場合にはこのセキュリティ機能を解除し、マクロ動作を有効にしなければならない。Fig. 2 (c)のように「このコンテンツを有効にする」にチェックすることで、本設計の自動画像挿入マクロが実行可能となる。なお、上記は本稿のPowerPointだけに特化したものではなく、信頼できるマクロを実行する場合の一般的な作業手順であり、WordやExcel使用時でも同様な表示や変更画面が出力されることを付記する。

(a)



(b)



(c)

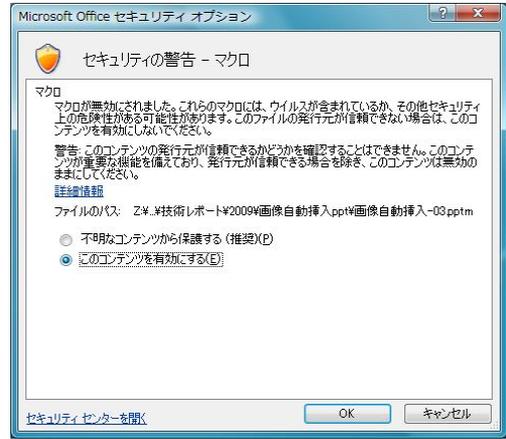
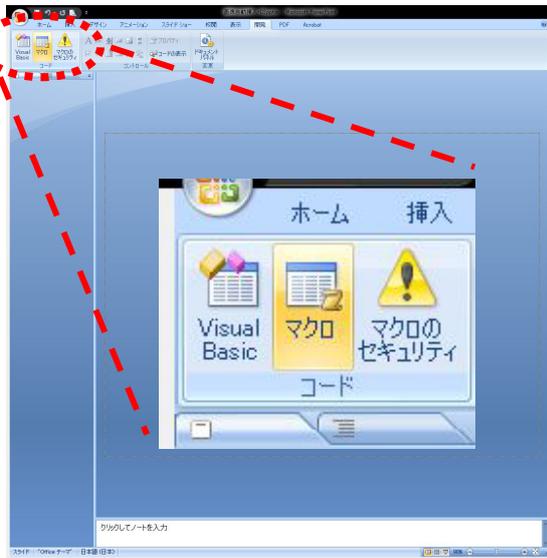


Fig. 2 PowerPoint 起動画面 (a)、マクロセキュリティ警告表示 (b) およびセキュリティオプションの変更画面 (c)

(a)



(b)

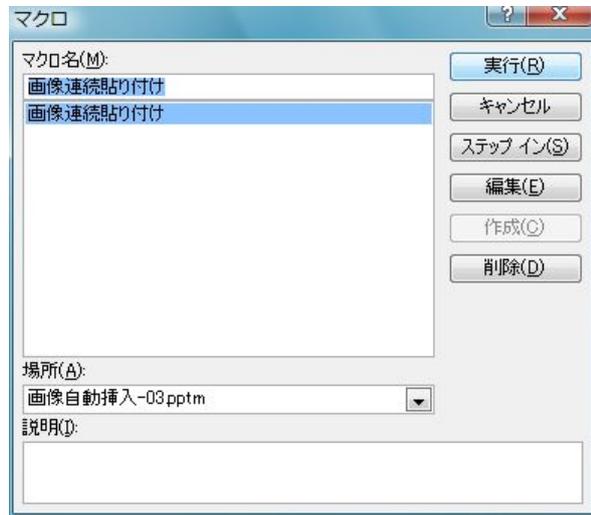


Fig. 3 マクロアイコン画面 (a) とマクロ選択/実行画面 (b)

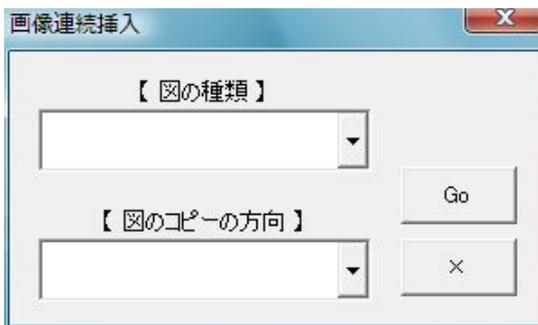
次に、メニューバーから「開発」タブに移動すると「マクロ」アイコンが現れる (Fig. 3 (a))。当該アイコンをクリックすると、登録されているマクロのリスト画面が出現し、所望のマクロを選択後、「実行」ボタンを押下するとマクロ処理が開始される (Fig. 3 (b))。本稿では、「画像連続貼り付け」というマクロ名が自動画像挿入機能を有するマクロのことである。

マクロを実行すると、Fig. 4 (a) に示すような自動画像挿入フォームが出現する。開発初期ということで、まずは「図の種類」と「図のコピーの方向」という 2 項目を作製した。こ

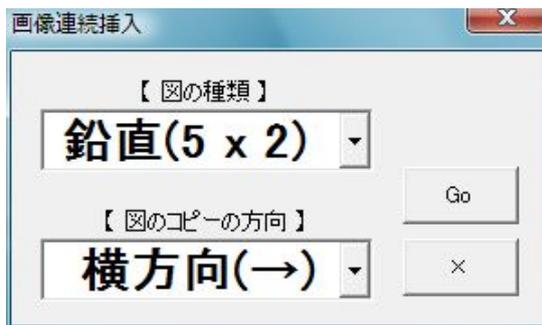
ここで「図の種類」とは、1枚のPowerPointスライドシート内に縦横何枚の写真/画像を挿入するかという設定項目である。また「図のコピーの方向」とは、「図の種類」で指定した縦横複数の写真/画像を縦方向/横方向のどちらに連続して貼付するかを指定する項目である。

例として、Fig. 4 (b) に示すように「図の種類」を「鉛直(5 x 2)」、「図のコピーの方向」を「横方向(→)」に設定した場合で説明する。この選択条件は、1枚のシート内に横方向に5列、縦方向に2行にわたって写真や画像を横方向に連続で挿入することを意味する。なお、図の種類「鉛直」という文言は作製上の都合によるもので、特段の意味はない。当然、貼付する枚数(X×Y)や連続貼付方向はカスタマイズ可能な変数である。同図(b)のように2項目を選択した後に「Go」ボタンを押すと、シートに貼付する画像ファイルの指定ウィザード(表示する拡張子は任意に設定可能)が出現する(Fig. 4 (c))。Fig. 4 (d) は、同図(c)において画像ファイルを11個選択した場合の結果である。図が小さくて見えにくいですが、横方向に5列、縦方向に2行の計10枚と次のスライドシートに1枚、即ち2シートに合計11枚の画像が自動でサイズ調整された状態で貼付されている。なお、隣接する画像同士のスペース幅や自動縮尺調整機能は本VBAマクロコード内に記述しており、前者に関連する詳細については後述する。

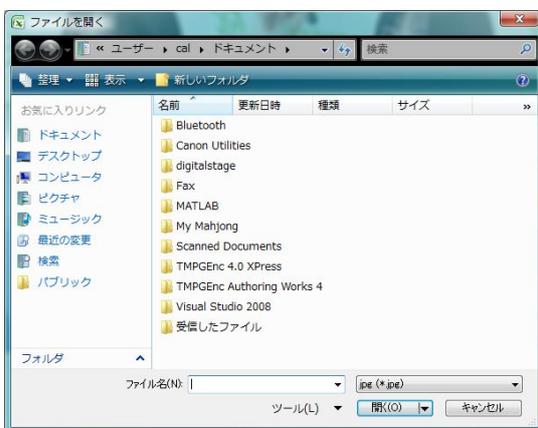
(a)



(b)



(c)



(d)

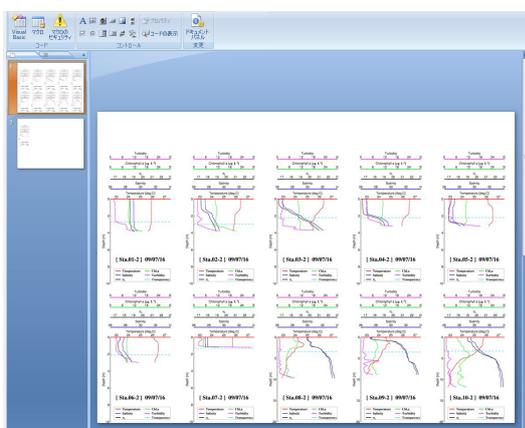


Fig. 4 「画像連続挿入」フォーム画面 (a)、条件選択した場合の一例 (b)、貼付画像選択画面 (c) および自動画像挿入マクロ実行結果の一例 (d)

次図は、図のコピー方向を横方向 (→) に設定した時 (Fig. 5 (a)) と縦方向 (↓) に設定した時 (Fig. 5 (b)) における画像の連続貼付結果を示す。画像内の Sta. 番号が連番になっているので、それを見ると分かりやすいが、それぞれの結果は破線の矢印に従った順に画像挿入されていることが確認できる。

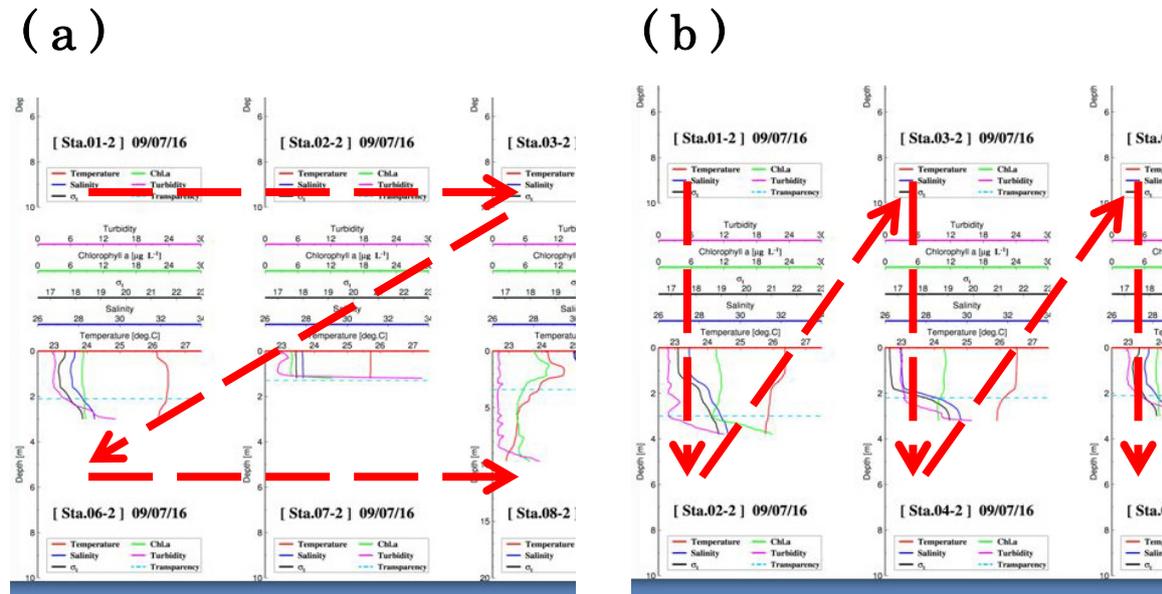


Fig. 5 図のコピー方向を横方向 (→) (a) と縦方向 (↓) (b) に設定した時の画像連続貼付結果の部分拡大図 (図の種類 : 5 x 2 設定時)

#### 4. 自動画像挿入マクロの拡張

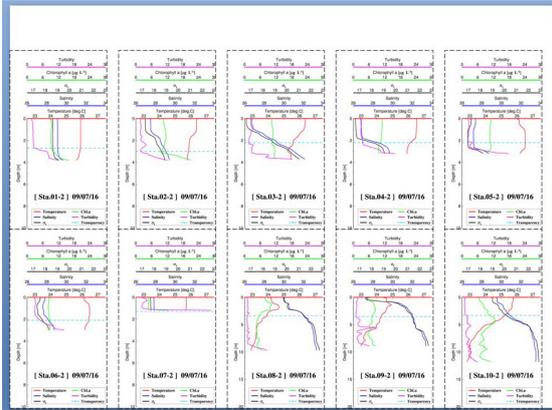
本章は、前章で説明しなかった設計の詳細について簡単に触れながら進めていく。前章では、スライドシート端と画像間の距離や隣接する画像同士の間隔といった PowerPoint シート内への画像貼付条件は、全て VBA コード内に直接記述していた。Fig. 4 (d) の各画像の外枠を破線で描画してみると Fig. 6 (a) のようになる。なお上記間隔幅の設定値は、以下の通りである (単位 : PPT 規定値)。

- 画像間 / 上下幅 : 0 、左右幅 : 0
- スライドシート端-画像間 / 上幅 : 60 、下幅 : 0 、左幅 : 0 、右幅 : 0

スライドシート端-画像間の上幅以外は設定値が全て「0」なので、スライドシート上には空隙なく画像が貼り付けられる設定なのだが、画像貼付時の自動縮尺調整機能 (縦横比を維持) の制約により、Fig. 6 (a) のように画像右端にスペースが生じてしまう場合がある。このような場合を除けば、設定値通りに正常動作することは確認済みである。例えば、上記の各幅設定値を全て 0 にした場合の出力結果を Fig. 6 (b) に示すが、当然、画像間スペースやスライドシート端-画像間スペースが全く生じずに画像が貼り付けられていることが確認できる。

もし、上述の設定条件や貼り付けパターン (X×Y) を変更したい場合、ユーザは毎回 VBA コードを修正/コンパイルした上で実行しなければならず、現状はあまりユーザフレンドリー

(a)



(b)

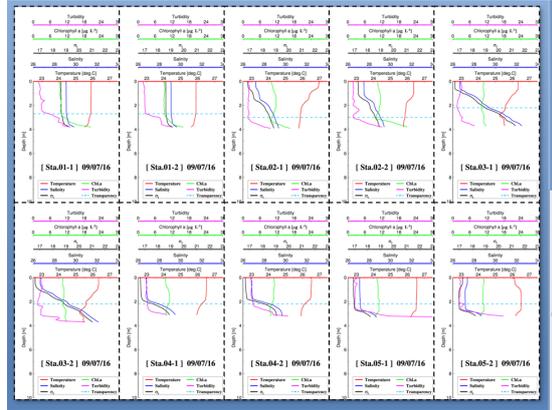


Fig. 6 画像貼付条件（スペース幅）の違いによるマクロ実行後の出力例

なソフトとは言い難い。そこで応用として、スライドシート端-画像間の設定幅を実行フォームから手動入力できるように新しくテキストボックスを配置した改良例を紹介する。当然やろうと思えば、画像間の上下/左右幅や画像貼付パターン（X×Y）も簡単に手動変更できるようにフォーム改良は可能であるが、これ以上のアップグレードは各人の努力に期待したい。

Fig. 7 (a) は、Fig. 4 (a) のフォーム上にスライドシート端-画像間の上下左右の設定幅を手動変更できるテキストボックスを配置した改良フォームである。図の種類を選択したら、スライド余白設定の各テキストボックス内に書き換え不可状態でデフォルト値が代入されるように設計した (Fig. 7 (a) では0を代入)。もし、手動で値を変更したい場合には真ん中のチェックボックスをクリックすることで各テキストボックスが書き換え可能状態になり、個別に所望の値を入力できるように配慮した (Fig. 7 (c))。

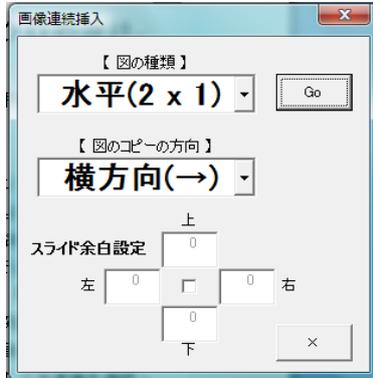
図の種類 (2 x 1) とコピー方向 (横方向→) を同一にした上で、スライド余白を Fig. 7 (a) および (c) のように設定した時のマクロ出力結果を、それぞれ Fig. 7 (b) および (d) に示す。なお、画像間の左右スペース幅は0としている。Fig. 7 (b) および (d) を見て分かるように、両者は各フォームで設定したスライド余白設定値通りに2つの画像が横並びに貼り付けられており、拡張機能が正常に動作していることが確認できる。

## 5. 結言

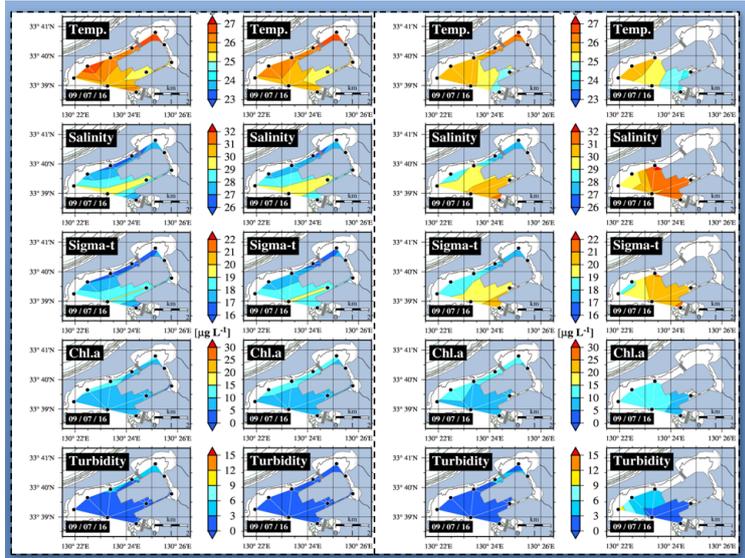
今回、VBAプログラミングによってMicrosoft PowerPoint 2007のスライドシート上に自動で画像挿入可能なVBAマクロの設計を行ない、実際の設計フォームや実行動作等について解説した。また応用として、本マクロを機能拡張した場合の一例も併せて紹介した。

VBプログラミングと聞くと、初心者には敷居が高いイメージを抱かれがちであるが、ユーザ設定値や諸動作を視覚的に判断・確認しながら作業できるメリットは非常に大きい。このVBAの長所を生かせるように基本的なことを少し勉強しさえすれば、本稿のようにある程度のことは自分の思い通りに設計できるようになり、ひいては現在各人が従事している作業の効率改善が図れる可能性も秘めている。本稿がVBAに取り組むきっかけとなれば幸いである。

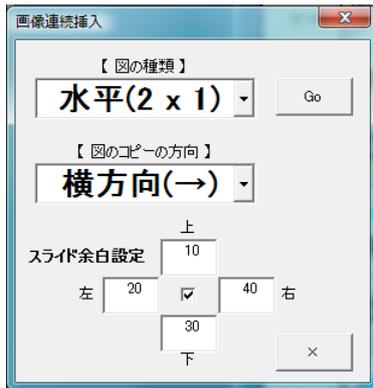
(a)



(b)



(c)



(d)

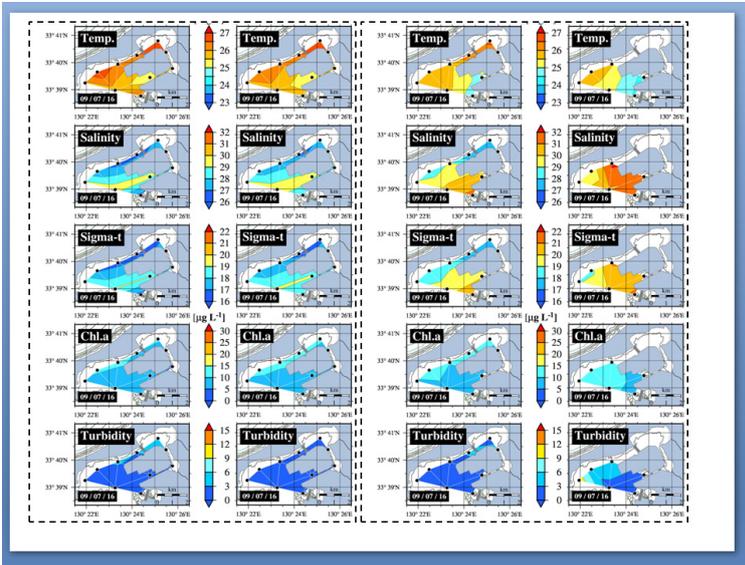


Fig. 7 画像貼付条件（スペース幅）の違いによる出力結果の例

### 参考文献

石井大輔 (2006) : 電子業務記録簿 (Excel 自動暦ツール) の作製. 九州大学応用力学研究所 技術職員技術レポート, 7, 91-98.

石井大輔 (2009) : Excel VBA 電子業務記録簿の改良とその設計事例. 九州大学応用力学研究所 技術職員技術レポート, 10, 7-24.

Excel VBA 実用サンプルコレクション, 渡辺ひかる, SOFTBANK Publishing (株), 2003

Excel 2003 VBA 辞典, (株) アンク, 翔泳社 (株), 2004