Introduction to IBM Datacap Taskmaster Capture 8.1 Development Kit

Robert Ferin
Advisory Software Engineer
IBM

Learn about the cross-industry and industry specific sample applications and tools available in the IBM Datacap Taskmaster Capture Development Kit (DDK), and find links to each of the components.

Overview

IBM Datacap Taskmaster Capture Version 8.1 automates input of data from documents to reduce cost and accelerate document process efficiencies. This advanced document capture solution helps you eliminate labor-intensive document preparation and manual data entry to streamline document processing and indexing. The Datacap Taskmaster Capture Development Kit (DDK) contains several different downloads that are useful to expanding your knowledge of Taskmaster and expanding the features within Taskmaster.

At a higher level, the sample applications we provide highlight features available in Datacap Taskmaster Capture (see Resources) that you can use for specific business needs. For example, if your type of business processing requires recognizing handwriting on a pre-printed form, keying from images, or taking advantage of other features, these scenarios are addressed by the sample applications. You can drop the samples into an existing Datacap Taskmaster Capture 8.1 installation.

If you want to expand or customize Datacap Taskmaster Capture, the DDK includes what is required to customize a DotEdit verification panel or a DotScan scan or fixup panel. With the custom panels, you can customize for specific forms or your specific vertical industry. And, with custom actions you can create your own actions, expanding the set of actions available to use in your Taskmaster applications.

Taskmaster provides web service access that allows other applications to integrate with it over http using a language independent mechanism. The wTM web service provides an interface that can create batches and access batch files and information.
**Target audience**

Software development kits are often targeted for those people with a strong technical knowledge. However, the software downloads available in this DDK appeal to two skill levels:

**Datacap Taskmaster Capture sample applications**
These apply to anyone with a working knowledge of running Datacap Taskmaster Capture applications.

**Datacap Taskmaster Capture custom actions, custom DotEdit Panels and custom DotScan Panels**
These apply to Datacap Taskmaster Capture application support staff who do custom implementations, reseller application support staff who do custom implementations, and customer IT staff responsible for Taskmaster Capture implementations.

**1040EZ (Finance)**

The 1040EZ sample application uses a hand written tax form as an example of just one vertical implementation.

The 1040EZ application shows examples of:

- Database lookup
- Database export
- Calculation and validation of numeric fields
- Handwriting recognition

See [Downloads](#) to access the 1040EZ sample application.

**1040EZ sample application installation**

Follow these steps to install the sample application:

1. Place the sample application into the Datacap directory. Typically this directory is C:\Datacap\1040EZ.
2. Edit the datacap.xml file and add a line for the 1040EZ application. If you install it into C:\Datacap, the new entry displays as: `<app name="1040EZ" ref="1040EZ"></app>`.
   - If 1040EZ is installed in a different directory, specify the full directory path in the `ref` parameter.
3. If 1040EZ is installed in a location other than C:\Datacap, configure the applicable directories for 1040EZ in the Application Manager.

The 1040EZ application is now ready to run.

**1040EZ application workflow**

The 1040EZ application consists of the following tasks:

- **VScan** – The virtual scan step that simulates the sample tax pages, read from the Images directory
• **PageID** – Identifies each page in the batch to confirm they are the expected tax forms
• **Profiler** – Aligns the page, recognizes, and validates the data on the page
• **Verify** – Displays each page to the operator to make any necessary corrections
• **Export** – Places the data from each page into an XML formatted file

### Handwriting recognition

The 1040EZ sample form contains handwritten data. The ICR/C engine works best on handwriting recognition. Before recognition, the scanned page is identified by matching fingerprints, preparing it for recognition. The recognition actions occur in the **Profiler** task.

### Database lookup

On this 1040EZ form, the name and address of a person are handwritten on the form. However, for this demonstration, this information is not obtained by recognition. The application looks up the name and address based on the social security number that is provided. At verify time, the image of the handwritten address is displayed along with the known information returned by the database. This display allows the verifier to compare the database information with the information provided on the form. It also reduces time that you might spend correcting recognition of this text.

The lookup rules run after recognition. The rule **TaxpayerName** runs an SQL query by using the action **ExecuteSQL**. The results of the query populate the field. Subsequent fields, such as **Address** and **City**, are populated by the **PopulateWithResult** action that takes the information from the previous query and stores it into the current field.

### Field validation and calculation

The Validate rule set has rules to validate each of the fields. For example, you can test the social security number (SSN) fields to confirm that the length is exactly nine characters by using the actions **IsFieldLengthMin(9)** and **IsFieldLengthMax(9)**. The spouse field has more validations that allow the field to be empty by checking for lengths of zero. The actions do not validate that the field contains only numbers because the fields in the Zones tab are configured to recognize as numbers, which also increases the likelihood that the text is recognized correctly. For example, the engine doesn't guess if a character is a numeric 1 or an alphabetic l, it defers to the numeric value.

The tax form has fields that are calculations of other fields, for example on the field **4AdjustedGross**, which calls **CalculateFields**. This action evaluates the expression and sets the status of the field that is based on the result. If the status fails, the user must review the fields to correct the error.

### Database export

The database rules operate at the batch level and at the page level. At the batch level, the database connection is opened by using the **ExportOpenConnection**. The database connection string is stored in the application server and is configured with the Application Manager. The smart
parameter @APPVAR(*/exportdb:cs) obtains the export database connection string. The sample string connects to an access database. To make the actions export to an SQL or Oracle database, the only change that is needed is to use the Application Manager to change the connection string to the database that you want.

It is best that you set passwords in the Application Manager by using a smart parameter. By doing so you encrypt passwords and prevent clear text passwords in the action parameters. Smart parameters are useful, be sure to review the documentation for them.

The data export occurs on the page level rule Rule: Page_1040ez. In this sample, the ExportFieldToColumn action is used to export the data for each page. As each action runs, the database query is created in memory and submitted to the database when the AddRecord action is done. After all the pages are processed, the rules that are attached to the batch close event are run, which calls Rule: 1040EZ_Close rule and the ExportCloseConnection.

1040EZ users and stations

Defined users
The following is the user information that is required for you to log in to the application. The admin ID has the full administrative permissions to view and modify the workflow settings of the application. The user IDs and passwords are case-sensitive.

<table>
<thead>
<tr>
<th>User ID</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>admin</td>
</tr>
<tr>
<td>edit1</td>
<td>admin</td>
</tr>
<tr>
<td>recog1</td>
<td>admin</td>
</tr>
<tr>
<td>scan1</td>
<td>admin</td>
</tr>
</tbody>
</table>

Defined stations
The following stations are predefined in the Survey application:
• 1
• 2
• local
• remote
• remote1
• remote2
• remote3

Express (Cross-industry)
The Express sample application uses a set of unorganized input forms to show examples of page separation, key from image, and double-blind data entry, where multiple operators verify the same documents and the results are compared. For example, this application might be used in a business scenario where the input documents are so varied, or for another business reason where it makes sense to hand key the data off the form, then uses an optional second user to key and verify the data. The verification interface is focused on the Datacap Taskmaster Capture web client.
This application shows examples of:

- Manual page identification
- Key from image (KFI)
- Two-user KFI verification
- Double-blind verification

See Downloads to access the Express sample application.

**Express sample application installation**

Follow these steps to install the sample application:

1. Place the sample application into the Datacap directory. Typically this directory is C:\Datacap \Express.
2. Edit the datacap.xml file and add a line for the Express application. If installed into C:\Datacap, the new entry displays as: `<app name="Express" ref="Express"></app>`. If Express is installed in a different directory, specify the full directory path in the `ref=` parameter.
3. If Express is installed in a location other than C:\Datacap, configure the applicable directories for Express in the Application Manager.

The Express application is now ready to run.

**Express application workflow**

The Express application is composed of the following workflows:

- **Remote Index** - Remote scan and upload with one-pass index data entry
- **Local All** - Local scanning with one-pass data entry
- **TwoPass** - Remote scan with two-pass data entry
- **Remote RScan** - Remote scan with one-pass data entry, requires a scanner
- **DoubleBlind** - Remote scan with double-blind, three-pass verification
- **Vscan to Upload** - Remote scan with one-pass data entry

**Defined shortcuts**

**Interactive**

- **Index** - Manual indexing, runs the Key tasks
- **RScan to Upload** - Remote RScan to upload
- **Upload** - Uploads images, uploads scanned images
- **Vscan to Index** - Remote VScan for indexing
- **Vscan Local** - Vscan on LAN

**Background**

As with any background task, the background tasks for Express can be run manually by using DotScan or you can configure Rulerunner to run them automatically.
• PrepData - Comparing and hiding data between passes  
• Assemble - Identifying pages by using barcode separator sheets and creating document structure  
• RRExport - Exports the data

Indexing

An indexing task is one that does not do recognition on the page. Also called key from image, or KFI, the operator manually enters field information by looking at the scanned image that is displayed on the screen. In the Express application, the operator selects the document type. Selecting the type causes fields to display on the screen that are appropriate for the image.

Two-pass verification

Two-pass verification requires two unique operators. The first operator manually enters the field data by viewing the on-screen image. After the verify operator submits the batch, it proceeds to the PrepData task and then becomes available for verification by a different operator. Because the setting on the Key2of2 step has the Queue by selection set to Other User, it requires a different operator to process the next verification step.

In the second verification step, the user sees the same images again. The user tabs through the fields, entering the data from the screen. If the second user’s entry matches the first user’s entry, the cursor moves to the next field. If the entry does not match, the field is erased and the cursor stays in the same field. The user then must enter the data again. This process repeats until the user enters the exact same data twice. After processing all the fields, the user can submit and move to the next page.

Double-blind verification / three-pass verification

The Express application also shows an example of full double-blind verification. In double-blind verification, three operators review the same documents. First, one operator performs KFI on the images in the batch. Next, a second operator also performs KFI on the same images in the batch. In this second step, the second operator’s entries are not compared to the first. After the batch is completed, it moves to a third operator.

The configuration of the Key3 step has the Queue by setting configured for Other Station And Other User. This setting requires a unique user and a unique station. During the indexing step, the job is not available for a user to select for verification unless the station and operator are different from the previous verifiers.

When the third operator performs the verification step, the fields between the first operator and second operator are compared. If the comparisons do not match, the field is shown in red. The third operator can review the failed fields and make corrections as necessary.

Express users and stations

Defined Users
The following is the user information that is required to log in to the application. The admin ID has the full administrative permissions to view and modify the workflow settings of the application. The user IDs and passwords are case-sensitive.

<table>
<thead>
<tr>
<th>User ID</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>admin</td>
</tr>
<tr>
<td>Indexer1</td>
<td>Indexer1</td>
</tr>
<tr>
<td>Indexer2</td>
<td>Indexer2</td>
</tr>
<tr>
<td>Indexer3</td>
<td>Indexer3</td>
</tr>
<tr>
<td>scan</td>
<td>scan</td>
</tr>
</tbody>
</table>

Two-pass and double-blind verification require multiple users. The same user cannot be the first and second verifier on the same batch. The administrative settings require unique users.

**Defined Stations**

The following stations are predefined in the application. Running the double-blind workflow requires the use of at least two of these stations:

- 1
- 2
- local
- remote
- remote1
- remote2
- remote3

**Walkthrough for two-pass operation in Express**

The Express application's multi-pass verification steps might be new to you if you are only familiar with the other sample and foundation applications that verify in a single step with one verifier, which is typical with recognition. To get you started with the Express application, the following is a walkthrough of the two-pass example provided by the Express application.

1. Log in to TMWeb as the first operator, Indexer1.
2. Select **Vscan to Index**.
3. Select **Two-pass**.
4. Virtual scan in images from the Express\Images directory.
5. Select **Upload**.
6. Run the **Assemble** task with DotScan or with Rulerunner.
7. In the TMWeb shortcut screen, select **Index**.
8. Review each page, ensure that the page is correctly identified by the barcodes, and fix any wrong types.
9. After a page has the correct type, enter the field information for that page.
10. After all pages are completed, submit the batch and log off.
11. Run **PrepData** task with DotScan or with Rulerunner.
12. Log in with a different user ID, such as Indexer2.
13. Select **Index**.
14. As each page is viewed, enter the data from the page. If the data matches the first operator, enter it once. If it does not match, enter it the same way twice for it to be accepted.
15. Submit the batch.
16. Use DotScan or Rulerunner to run the **RRExport** step.
17. The export task creates a flat text file in the Express\Export directory.

**Survey (Cross-industry)**

The Survey sample application uses a sample survey form to collect handwritten information and user selections on the form. The process shows the use of a dropout form, optical mark recognition (OMR), page ID with barcodes, and fixed pitch handwriting recognition with Taskmaster Capture recognition. Anchors are used in this application to properly align the scanned image with the existing fingerprint, which provides better recognition and reduces manual error correction.

An example can be applied to your vertical application. Dropout colors on pre-printed forms provide for better recognition and OMR detection.

This application shows examples of:

- Recognition of fixed pitch handwritten text
- Using a dropout color form
- Optical mark recognition (OMR)
- Using barcodes that are printed on the form (no separator pages) for page identification
- Using anchors for page alignment to a fingerprint

See [Downloads](#) to access the Survey sample application.

**Survey sample application installation**

Follow these steps to install the sample application:

1. Place the sample application into the Datacap directory. Typically this directory is C:\Datacap \Survey.
2. Edit the datacap.xml file and add a line for the Survey application. If installed into C:\Datacap, the new entry displays: `<app name="Survey" ref="Survey"></app>`.
   If Survey is installed in a different directory, specify the full directory path in the `ref=` parameter.
3. If Survey is installed in a location other than C:\Datacap, configure the applicable directories for Survey in the Application Manager.

The Survey application is now ready to run.

**Survey application workflow**

The Survey application is composed of the following tasks:

- **VScan** – The virtual scan step that simulates the scanning of four sample pages, read from the Images directory
- **PageID** – Identifies each page in the batch to confirm they are the expected Survey forms
- **Profiler** – Performs aligns the page, recognition, and validates the data on the page
- **Verify** – Displays each page to the operator to make any necessary corrections
- **Export** – Places the data from each page into an XML formatted file

### Using dropout forms

The file Survey Blue Dropout.PDF shows a color form that uses a drop-out color. The color can change based on your scanner. Dropout forms allow easy removal of the form lines before recognition, providing better character recognition and OMR recognition results. The original source input form is in the Examples directory. There are two different example forms.

A dropout form can be used with a scanner which removes the dropout color during scanning, leaving just the text to be recognized on the image within the batch. Color dropout is handy because the scanner does the work and it leaves the hand written information untouched.

For forms that must be black and white, there is a sample form that uses light gray dotted lines. These dotted lines can be removed by the ImageEnhance action with the despeckle settings or use tolerance settings to remove all light gray information. Similarly, thin solid lines can be used with the line removal settings. When you use black and white forms, be careful that you retain the ability to remove lines without removing the hand written information. Black and white line removal allows forms to be faxed or copied.

### Zoning a dropout form

In the Examples directory you can find the file Fingerprint Creation Template.tif. In this form, all of the dropout information is gone and there are marks that indicate each zone. This image shows a handy way to create the zones for your form in Datacap Studio. To create do so, mark the opposite corners for each zone on your original paper form to show the location and size. Scan the form and run it through your dropout process. The result is an image with the dropout information gone but your zone marks remain. Use this image as your fingerprint and then zone your fields by using your zone reference marks.

### Barcode page identification

The Survey sample form contains a barcode for page identification, which is an alternative method of page ID. No fingerprint matching is required when barcodes are used. If you can control the format of your form and if you have multiple form types, you might want to consider using a barcode for page identification. In this example, the PageID rule set looks for the barcode, locates it, and sets the fingerprint without doing fingerprint matching.

### Anchors for form alignment

The form has a "T" shaped line at the top and another at the bottom. These areas are zoned as a field and additionally designated as **Anchor fields**. Before recognition, the **MatchPattern** action is used to align the scanned form to the fingerprint by using the anchors. This registration allows better recognition, OMR detection, and viewing during verification, even if you are also using a deskew setting during image enhancement. One or more anchors can be used to align an image, usually two or three anchors work best for alignment. It is not required that the anchors be at the...
top and bottom, as in this example. You can have a registration anchor on opposite corners of the page.

A single anchor can also be used for registration. Two or more anchors are useful as they can compensate better for skewing or stretching of the image.

**Optical mark recognition (OMR)**

Parts of the sample form have OMR fields. OMR fields can be recognized with the action `RecogOMRThreshold`. The action tests the OMR field to determine whether there is a deliberate mark in the area. OMR detection works best with dropout forms, which are what Survey uses. You can view the recognition rules to see how OMR actions are set up. The parameters are adjusted based on the size of the OMR box.

With black forms (pages that have no dropout or line removal), OMR detection can still be performed. You must adjust the action input parameters to ignore the black on the printed form. Another tip for black forms where the OMR field is a black line box that is filled with a stroke or mark, is to create the zone around the whole box black line box, so the black form box is included in the zoned field. Set the OMR action detection parameters to ignore the black lines from the box. Including the whole box in the zone helps reduce OMR detection errors if the form does not exactly align, which can cause part of the black line box to be considered a mark.

**Handwriting recognition**

The ICR/C engine, which is best for doing handwriting recognition, performs the field recognition. The written text on the form is fixed width, so the zones for each field are configured in the Zones tab to indicate that they are fixed pitch handwriting. After verification, the application will create an export file in an XML format.

**Validation**

The Validate rule set performs a number of tests. Some OMR groups can have multiple selections and others can have a specific number only. These validations are performed with the actions `IsMinOMRChecked` and `IsMaxOMRChecked`. For example, only one payment type can be checked. Multiple contributions can be checked and the form must be signed.

The form also validates dates and amounts. For example, the start date must be before the end date. The total contribution must match the calculation that is based on the frequency. It is possible to prevent an operator from passing a validation that fails, this application allows a document to go through that has validation errors.

**Survey users and stations**

**Defined users**

The following is the user information that is required to log in to the application. The admin ID has the full administrative permissions to view and modify the workflow settings of the application. The user IDs and passwords are case-sensitive.
<table>
<thead>
<tr>
<th>User ID</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>admin</td>
</tr>
<tr>
<td>edit1</td>
<td>admin</td>
</tr>
<tr>
<td>recog1</td>
<td>admin</td>
</tr>
<tr>
<td>scan1</td>
<td>admin</td>
</tr>
</tbody>
</table>

**Defined stations**

The following stations are predefined in the Express application.

- 1
- 2
- 3
- 4
- background
- remote1

**Custom DotEdit panels (Cross-industry)**

Taskmaster Capture provides the DotEdit client application for verification of document data and metadata. You can use Microsoft® Visual Studio® 2010 to create custom DotEdit verification panels for Datacap Taskmaster Capture applications.

Custom panels open up the Datacap Taskmaster Capture application to allow full customization of the verification panel to match your specific needs. The download link provides the foundation source code that is needed to create and modify a panel.

Custom DotEdit verification panels provide:

- Control for the layout of fields and images
- Access to the Datacap Object (DCO) and the application variables
- Custom functions specific to your vertical application
- Display of custom logos or corporate images on your panel

See [Downloads](#) to access the custom DotEdit panels. TM-DDK-8.1-FP2-DotEditPanels.zip. The deprecated download, Taskmaster-DDK-8.1-DotEditPanels.zip, can be continued to be used for Taskmaster 8.1 and 8.1 FP1.

For more information about how to build data verification panels for DotEdit, consult the [Creating Custom DotEdit Panels Guide](#).

When using the toolkit, you must download the version that matches the current version of your software. If you are using FP3, then you must use the FP3 version of the toolkit to build your panels. If you are moving from FP2 to FP3, then you must download the FP3 compatible version and use it to rebuild your panels that were built for FP2. If you are still using IBM Datacap Taskmaster Capture 8.1 fix pack 2. In order to compile the panels, 8.1 FP2 needs to be installed on your development machine, and you will need an additional DotEdit update available as a hot fix: 8.1-Datacap-Taskmaster-FP2-DotEditDotScan-HotFix.zip. This update is available on [IBM Fix Central](#).
A second update of the DotEdit panel package has been made to include the *Universal Field-At-A-Time* panel. While the generic Universal panel can be customized for your applications, an update to DotEdit is required to enable customization of the Universal panel. This update is available on IBM Fix Central, and is different from the previous update mentioned above. Download the file TM-DDK-8.1-FP2-DotEditPanels-with-Universal.zip from the downloads section, to customize the Universal panel. The download package contains instructions for customizing the panel and information regarding the required update to DotEdit to be used with version 8.1 FP2.

**Custom DotScan Panels (Cross-industry)**

Taskmaster Capture provides the DotScan rich client application for scanning paper documents, selecting and processing existing document files, and manually running rules-based workflow tasks. You can use Microsoft Visual Studio 2010 to create custom DotScan panels for Taskmaster applications.

Custom panels open up the Taskmaster application to allow customization of DotScan panels to match your specific needs. The download link provides the foundation source code that is needed to create and modify a panel.

Custom DotScan panels provide:

- Control for the layout of the scanning UI
- Access to the XML representation of the Batch and the Document Hierarchy (Setup DCO)
- Custom functions specific to your vertical application
- Display custom logos or corporate images on your panel
- Access to a controlled and supported subset of ISIS and TWAIN scanning capabilities that are contained in the dcScanISIS and dcScanTWAIN components

To successfully design and build custom DotScan panels requires familiarity with:

- C# programming, the Microsoft .NET framework, and Windows Forms
- Microsoft Visual Studio
- XML, Microsoft System.XML namespace, and XPath syntax
- Taskmaster Document Hierarchy (Setup DCO) and the structure and contents of batches in Taskmaster

**The DotScan Panels package**

By using the DotScan project that is available to download you can customize DotScan as needed for your Taskmaster application. See the developer document within the download package that describes how to create new panels or modify existing panels for scanning, batch creation, fixup, or other batch-oriented tasks. The document also illustrates how different functional elements (image view, batch view, metadata fields, and more) are implemented and manipulated within the panels. Microsoft Visual Studio 2010 is required to create or modify DotScan panels.

When using the toolkit, you must download the version that matches the current version of your software. If you are using FP3, then you must use the FP3 version of the toolkit to build...
your panels. If you are moving from FP2 to FP3, then you must download the FP3 compatible version and use it to rebuild your panels that were built for FP2. If you are still using IBM Datacap Taskmaster Capture 8.1 fix pack 2. In order to compile the panels, 8.1 FP2 needs to be installed on your development machine, and you will need an additional DotScan update available as a hot fix: 8.1-Datacap-Taskmaster-FP2-DotEditDotScan-HotFix.zip. This update is available on IBM Fix Central.

See Downloads to access the custom DotScan Panels project in TM-DDK-8.1-FP2-DotScanPanels.zip. The deprecated panels in DotScanPanelsDownload.zip can continue to be used for Taskmaster 8.1 and 8.1 FP1.

**Custom actions**

For situations where a specific application need cannot be satisfied with standard actions, custom actions can fulfill virtually any need. You can use Microsoft Visual Studio 2010 to create custom actions for your Taskmaster applications. Action libraries are written in C# .NET, implemented as DLLs, and made available to your Taskmaster applications through a Datacap RRX file that describes the library's public interface. The actions that you create are added to your Taskmaster installation, making them available for use within your Taskmaster applications. Custom actions display in Datacap Studio allowing them to be added to rule sets in the same manner as standard actions.

The action DLL templates contained in TM-DDK-8.1-NET-ActionTemplates.zip are Visual Studio templates that create the base structure for an action DLL and contain sample code that demonstrate how to access the common internal methods used in an action DLL. These DLLs embed the RRX file within the DLL itself, so only the DLL needs to be deployed to the target install. The action DLL does not require registration, as was required by the older style template. The zip contains two templates: a 2.0 template that uses .NET 2.0 and a 4.0 template that uses .NET 4.0 — either of which can be used to create custom actions.

The deprecated TM8.1-CustomActions.zip download package contains the Visual Studio C# template to create your own actions. These old-style actions continue to be supported in Taskmaster 8.1. It is recommended that you use the newer-action template mentioned above for any new actions because it is easier to manage and deploy these actions. The old style requires both the RRX and DLL to be deployed, and the DLL must be registered. Also included in this .zip file are two old-style sample action library implementations and a sample application to run the sample actions. See Downloads to access the custom actions template and sample actions.

Find documentation for the custom actions template, the sample actions, and details to use these actions in an application in the IBM Datacap Taskmaster Capture 8.1 documentation.

**wTM web service**

With the wTM web service you can integrate external applications with Taskmaster Capture solutions. With wTM you can create Taskmaster batches, upload images, and retrieve captured data and processed images. To create solutions that use wTM you must be familiar with calling
web services in your chosen programming language such as a C#. If including meta-data when creating a Taskmaster batch, or retrieving captured field data, you must also be familiar with the Taskmaster batch structure is also required.

See Downloads to access the wTM sample code projects and documentation. The download package contains:

- C# sample code demonstrating the calling of each endpoint
- A Microsoft® Windows Forms application demonstrating calling endpoints in sequence to perform common tasks such as batch creation and retrieving captured data
- Installation instructions, instructions for using the included client application, and concepts for working with wTM

**Conclusion**

This article described the IBM Datacap Taskmaster Capture Developer Kit and how it can enhance your product experience. The sample applications highlight some of the features available and how you can use them in your own application. For greater control of the verification screen, I included information about customization of the DotEdit verify panel, which allows full control of the panel that is presented by your application. By using custom actions you can expand the abilities available for your applications.
## Downloads

<table>
<thead>
<tr>
<th>Description</th>
<th>Name</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1040EZ sample application</td>
<td>Taskmaster-DDK-8.1-1040ez.zip</td>
<td>522KB</td>
</tr>
<tr>
<td>Express sample application</td>
<td>Taskmaster-DDK-8.1-Express.zip</td>
<td>380KB</td>
</tr>
<tr>
<td>Survey sample application</td>
<td>Taskmaster-DDK-8.1-Survey.zip</td>
<td>757KB</td>
</tr>
<tr>
<td>DotEdit sample</td>
<td>Taskmaster-DDK-8.1-DotEditPanels.zip</td>
<td>54KB</td>
</tr>
<tr>
<td>DotEdit Panels for 8.1 FP3</td>
<td>TM-DDK-8.1-FP3-DotEditPanels-with-Universal.zip</td>
<td>837KB</td>
</tr>
<tr>
<td>DotScan Panels for 8.1 FP3</td>
<td>TM-DDK-8.1-FP3-DotScanPanels.zip</td>
<td>837KB</td>
</tr>
<tr>
<td>DotEdit Panels for 8.1 FP2</td>
<td>TM-DDK-8.1-FP2-DotEditPanels.zip</td>
<td>233KB</td>
</tr>
<tr>
<td>DotEdit Panels for 8.1 FP2 with Universal panel</td>
<td>TM-DDK-8.1-FP2-DotEditPanels-with-Universal.zip</td>
<td>233KB</td>
</tr>
<tr>
<td>DotScan Panels project</td>
<td>DotScanPanelsDownload.zip</td>
<td>587KB</td>
</tr>
<tr>
<td>DotScan Panels for 8.1 FP2</td>
<td>TM-DDK-8.1-FP2-DotScanPanels.zip</td>
<td>1333KB</td>
</tr>
<tr>
<td>Custom actions sample</td>
<td>TM8.1-CustomActions.zip</td>
<td>1038KB</td>
</tr>
<tr>
<td>Custom actions .NET 2.0/4.0</td>
<td>TM-DDK-8.1-NET-ActionTemplates.zip</td>
<td>578KB</td>
</tr>
<tr>
<td>wTM sample</td>
<td>wTMSample.zip</td>
<td>346KB</td>
</tr>
</tbody>
</table>
Resources

- Access the **IBM Datacap 9.0 Developer Kit**.
- **IBM Datacap Taskmaster Capture**: Find more information on the product page.
- **developerWorks on Twitter**: Join today to follow developerWorks tweets.
- **developerWorks podcasts**: Listen to interesting interviews and discussions for software developers.
- **developerWorks community**: Connect with other developerWorks users while exploring the developer-driven blogs, forums, groups, and wikis.
- **Datacap Technical Mastery Business Partner Community**: See what Taskmaster talk is happening among other developers and get involved.
About the author

Robert Ferin

Robert Ferin is a developer for IBM Datacap Taskmaster Capture. He has been with the team since 2007, recently focusing on the localization support of the Taskmaster product. Prior to that time, he has accumulated 20 years of software development experience.

© Copyright IBM Corporation 2012, 2015
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)