Microsoft Dynamics GP

Development Tools
for Microsoft Dynamics GP

White Paper

Date: March 2006

http://www.microsoft.com/dynamics/gp/default.mspx
Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship and supply chain processes in a way that helps you drive business success.

U.S. and Canada Toll Free 1-888-477-7989
Worldwide +1-701-281-6500
www.microsoft.com/dynamics
Development Tools for Microsoft Dynamics™ GP

Summary: Several development tools allow developers to create custom applications that integrate with Microsoft Dynamics GP. If you are a developer working for a Microsoft® independent software vendor (ISV), you can use these tools to integrate an existing application with Microsoft Dynamics GP or to develop customized applications for vertical markets.

Contents
- Overview
  - Introduction to Microsoft Dynamics GP
  - Intended Audience
  - Architecture
- Developer Toolkit for Microsoft Dynamics GP
- Choosing a Tool
  - Developing a New Application
  - Customizing Windows
  - Enabling Real-Time Integration
  - Completing a Batch Integration
- Understanding the Tools
  - eConnect
  - Integration Manager
  - Open Database Connectivity (ODBC) Drivers
  - Modifier with Microsoft Visual Basic® for Applications (VBA)
  - Extender
  - Dexterity®
  - Microsoft Dynamics GP Developer Resources documentation
  - Continuum Application Programming Interface (API)
  - Business Portal SDK
- Reporting Tools

Overview
This article briefly describes Microsoft Dynamics GP architecture, summarizes the features and benefits of each Microsoft Dynamics GP development tool, and provides guidelines for choosing the right tools for various development tasks. The toolset for Microsoft Dynamics GP may vary depending on your status as an ISV or customer; therefore, you might not have access to all of the tools discussed in this article.
Several tools are available for developers to create custom applications that integrate with Microsoft Dynamics GP. These tools include the following:

- eConnect
- Integration Manager
- ODBC Drivers
- Modifier with VBA
- Extender
- Dexterity
- Microsoft Dynamics GP Developer Resources documentation
- Continuum API
- Business Portal SDK

You can obtain these development tools from the Microsoft Dynamics GP CDs, or if you’re an MSDN® Universal subscriber, you can download Microsoft Dynamics GP CD images from the MSDN subscriber downloads area. See the MSDN Subscriptions site for more information about MSDN subscriptions.

Dexterity and the Continuum API also can be downloaded from the PartnerSource extranet. You must use Microsoft Passport Network authentication to access the extranet.

**Introduction to Microsoft Dynamics GP**

Microsoft Dynamics GP is a comprehensive accounting and business management system. It supports business processes in the lower mid-market and scales up to meet the requirements of complex business processes in the upper mid-market and corporate market.

Microsoft Dynamics GP divides business processes into modules, which are collections of applications that work together to meet a number of closely related accounting and business management goals. Modules that have similar purposes are organized into solution series.

Using the tools described in this article, developers can customize various Microsoft Dynamics GP modules for their customers or write applications that integrate with Microsoft Dynamics GP.

For more information about the functionality of Microsoft Dynamics GP, visit the Microsoft Dynamics GP Web site.

**Intended Audience**

This technical article is for software engineers who develop applications that are designed to work with Microsoft Dynamics GP. The topics in this article are useful for ISVs and other developers who might customize or integrate with Microsoft Dynamics GP. Readers should have basic experience with programming for business applications and using data models.
Different tools require different skill sets. To perform simple integrations, such as importing data from a table or a Microsoft Office Excel® file, you need experience with Microsoft Office applications, and experience with VBScript is recommended. For many types of customization, experience with automation, project management, and development significantly aids in project completion. For the most complex projects, you need advanced programming skills and knowledge of the networks and operating systems supported by Microsoft; you should be familiar with the business logic implementation of your application and with extensible markup language (XML) coding, XML schemas, Structured Query Language (SQL) programming and procedures, and scripting languages, such as Microsoft Visual Basic.

Architecture

Microsoft Dynamics GP was designed with an open architecture, which means that there are a number of ways that the application can be customized and integrated. At a high level, the architecture can be broken down into four basic components: the database layer, the data dictionary layer, the business logic layer, and the user interface layer.

Integrating applications can interact with Microsoft Dynamics GP at any of these four layers. Most integration should occur at the business logic layer, which ensures valid data and the best performance. Integrating to the database layer is complex and introduces the possibility that data will not be validated; modifying the user interface layer can reduce performance.

In the following diagram, the four layers of architecture are illustrated in the "Microsoft Dynamics GP Architecture" column. Notice the close relationship between the business logic layer and the data dictionary layer; this is because the data dictionary defines the business logic rules. Pointing to each layer in the architecture are tools that you can use to integrate your external application with Microsoft Dynamics GP or to customize Microsoft Dynamics GP. These tools are described later in this article.

Figure 1. Microsoft Dynamics GP architecture

Database Layer

Database-level integration allows direct access to the data and is supported by many development environments. However, this is the most difficult layer for integration because all the data must be validated. The developer needs to understand what tables
need to be populated, what fields are required, and how to ensure that the data is error-free. Tools that can integrate at this layer include Dexterity and the Continuum API. Using the Microsoft Dynamics GP Developer Resources documentation as a reference, you also can use ODBC drivers for Microsoft SQL Server™ to access the database layer directly.

eConnect also integrates at this layer, but unlike the other tools, eConnect enforces business logic.

Data Dictionary Layer

The data dictionary layer contains all of the resources used to define and create the Microsoft Dynamics GP data model in Microsoft SQL Server. These resources are built within the Dexterity toolset and include data types, fields, and tables.

Resources for foundational and financial components are stored in the main data dictionary. Some components are developed as integrating applications and have their own data dictionaries. These include manufacturing, project management, and human resources components. Likewise, applications developed in Dexterity by ISVs are developed as separate integrating applications and have their own data dictionaries.

In a multidictionary environment, the runtime engine interprets the main dictionary and the integrating dictionaries at the same time. This capability allows multiple integrating dictionaries to function with the main dictionary, providing a seamless user experience.

The only tool that allows integration at the data dictionary layer is Dexterity. With Dexterity, you can read tables, create new tables, and add new or existing fields to those tables. ISVs cannot add fields to the existing tables within the Microsoft Dynamics GP data model. This method ensures the integrity of the Microsoft Dynamics GP data model for all ISVs.

Business Logic Layer

The business logic layer allows you to take advantage of the internal code that is shipped with Microsoft Dynamics GP, so you do not need to spend time developing code that already exists.

Three tools provide the primary means for accessing the business logic: eConnect, Dexterity, and the Continuum API. eConnect uses SQL stored procedures to follow the business logic rules and to ensure that data is valid in Microsoft Dynamics GP, while ensuring good performance. Dexterity gives you access to hundreds of procedures used by the application, such as General Ledger posting, in addition to all of the table definitions. The Continuum API uses Microsoft Component Object Model (COM) Automation, which allows non-Dexterity applications to call a Dexterity procedure.

In addition to these three tools, the SQL Optimized, Fixed Assets Management, and Project Accounting destination adapters for Integration Manager also support some integration at the business logic layer.

User Interface Layer

The last level for integration is the user interface. Of course, data can be entered manually into the windows of Microsoft Dynamics GP. Integration Manager’s Great Plains adapter integrates data through the user interface. You also can use the
Continuum API, Modifier with VBA, or Dexterity to integrate at this level by automating the windows.

**Developer Toolkit for Microsoft Dynamics GP**

The Developer Toolkit for Microsoft Dynamics GP is expected to become available in April 2006. The Developer Toolkit will include documentation, samples, and command-line tools and compilers to help developers create and deploy .NET Framework applications that can help you to connect external applications to financial management data in Microsoft Dynamics GP.

The Developer Toolkit will include the following components.

- eConnect SDK
- eConnect Runtime
- Web Services for Microsoft Dynamics GP SDK
- Web Services for Microsoft Dynamics GP Runtime
- Visual Studio® Tools for Microsoft Dynamics GP SDK
- Visual Studio Tools for Microsoft Dynamics Runtime

You will be able to use Visual Studio Tools for Microsoft Dynamics GP to create application user interfaces that are consistent with Microsoft Dynamics GP. Use Web Services to integrate those applications with Microsoft Dynamics GP utilizing the business logic within eConnect.

See [eConnect](#) for more information about the eConnect SDK and eConnect Runtime.

**Choosing a Tool**

If you're a developer who is customizing or developing integrating applications with Microsoft Dynamics GP, the following tools are available:

### Integration

These tools allow you to transfer data from one source to another.

- [eConnect](#)
- Integration Manager
- Extender
- ODBC Drivers
- Dexterity

### Development and Customization

These tools allow you to create new applications that are based on Microsoft Dynamics GP or to customize existing applications to fit specific needs.

- Dexterity
- Modifier with VBA
- Extender
- **Business Portal SDK**
Documentation These tools provide information about integrating with the Microsoft Dynamics GP data model and typically are used in conjunction with other Microsoft Dynamics GP development tools.

- Continuum API
- Microsoft Dynamics GP Developer Resources documentation
- Business Portal SDK

To help you determine which development tools to use, this section describes some common tasks that you may want to perform and defines the tools most appropriate for each task. When multiple tools are listed for a task, the order that the tools are listed in is not a recommendation about the ranking of the tools for the task, except where noted.

The following topics are discussed:

- Developing a New Application
- Customizing Windows
- Enabling Real-Time Integration
- Completing a Batch Integration

For more information about each tool, see Understanding the Tools later in this article.

Developing a New Application

You can develop a new application that integrates with Microsoft Dynamics GP to meet the needs of a particular industry or a vertical market. In this scenario, you are either starting to build a new application or rewriting an existing application.

Table 1.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build a new application or rewrite an existing application, and then perform real-time or batch integration to Microsoft Dynamics GP.</td>
<td>Visual Studio .NET to write the application, and then eConnect for real-time and batch integration, or Integration Manager for batch integration</td>
</tr>
<tr>
<td>Create a rich-client, seamless integration with Microsoft Dynamics GP.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>Create a new business component, query page, Web Part, or portal page for Business Portal to integrate with Microsoft Dynamics GP.</td>
<td>Visual Studio .NET, Rational XDE and its code generator, the Entity Relational Map Generator, and the Entity Association Map Generator to write the application, and then the Business Portal SDK for integration.</td>
</tr>
</tbody>
</table>

Customizing Windows

You can modify the windows in the Microsoft Dynamics GP application to meet the specific needs of an industry, vertical market, or individual end user.
The following topics are discussed:

- **Rearranging Fields**
- **Adding Fields**
- **Changing the Properties of a Field**
- **Changing Text Labels**
- **Customizing the Business Logic**

### Rearranging Fields

You can rearrange the fields in an existing Microsoft Dynamics GP window, such as the Customer Maintenance window or the Receivables Transaction Entry window. Fields can be rearranged to meet the needs of the customers who use your integrating application and Microsoft Dynamics GP for a certain industry or vertical market.

**Table 2.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make this change for only one end user or a small number of end users, or for multiple end users who have different requirements.</td>
<td><strong>Modifier with VBA</strong></td>
</tr>
<tr>
<td>Make this change for all or most of the end users.</td>
<td><strong>Dexterity</strong></td>
</tr>
</tbody>
</table>

**Note**  If the existing window or field resides in an integrating dictionary, use **Modifier with VBA** rather than **Dexterity**.

### Adding Fields

You can add new fields and store additional data in an existing Microsoft Dynamics GP window. For example, you could track additional customer attributes in the Customer Maintenance window.

**Table 3.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make this change for only one end user or a small number of end users.</td>
<td><strong>Modifier with VBA</strong> can add the field to the window, and VBA provides the business logic to keep the data synchronized with other Microsoft Dynamics GP data and windows.</td>
</tr>
</tbody>
</table>
For This Task | Use This Tool
--- | ---
Add these fields for all or most of the end users. | **Dexterity** enables you to create custom lookups for the new fields. **Extender** can be used to create a new window that is linked to an existing window. Fields in the new window can be used to store additional data that is related to records entered in the existing window.

**Note** If the existing window or field resides in an integrating dictionary, use **Modifier with VBA** rather than **Dexterity**.

### Changing the Properties of a Field

You can change the default values of a field's properties. For example, you could make the field a required field or hide the field from view.

**Table 4.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make this change for only one end user or for a small number of end users.</td>
<td><strong>Modifier with VBA</strong></td>
</tr>
<tr>
<td>Make these property changes, in addition to other changes to windows, for all or most of the end users.</td>
<td><strong>Dexterity</strong></td>
</tr>
<tr>
<td>Make only these property changes, for all or most of the end users.</td>
<td><strong>Continuum API</strong></td>
</tr>
</tbody>
</table>

**Note** If the existing window or field resides in an integrating dictionary, use **Modifier with VBA** rather than **Dexterity**.

### Changing Text Labels

You can change the text labels on an existing window to meet the needs of a specific industry or a specific customer. For example, in a medical setting you could change all occurrences of the term "Customer" on Microsoft Dynamics GP windows to "Patient."

**Table 5.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make one change to a text value and have this change affect all other instances of this text appearing on other windows within Microsoft Dynamics GP.</td>
<td><strong>Modifier with VBA</strong></td>
</tr>
<tr>
<td>Make changes to each window individually.</td>
<td><strong>Modifier with VBA</strong> or <strong>Dexterity</strong></td>
</tr>
</tbody>
</table>

**Note** If the existing window or field resides in an integrating dictionary, use **Modifier with VBA** rather than **Dexterity**.
Customizing the Business Logic

You have several options for adding business logic or exerting control over the business logic running in an existing window. You may simply want to insert default data into a field as the end user opens a window or presses the TAB key to move out of a field. In a more complex scenario, you might want to stop a business process, such as posting an accounts receivable transaction, based on a certain set of criteria.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make this change for only one end user or a small subset of end users.</td>
<td>Modifier with VBA</td>
</tr>
<tr>
<td>Make business logic changes for all or most of the end users, in addition to user-interface changes to existing windows.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>Make simple business logic changes using another COM-compliant development tool for all or most of the end users, but no user-interface changes to existing windows.</td>
<td>Continuum API</td>
</tr>
</tbody>
</table>

Enabling Real-Time Integration

Some tools allow integrations in which data is written to and read from Microsoft Dynamics GP in real time. This means that you can continually update the Microsoft Dynamics GP databases while new data is entered into your systems.

ISVs typically should build an application using Microsoft Visual Studio .NET and connect the application using eConnect. In certain circumstances, the other tools might be a better choice.

The following topics are discussed:

- [Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)]
- [Subscribing to Changes within Microsoft Dynamics GP Data]
- [Writing Simple Data to Microsoft Dynamics GP]
- [Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)]
- [Writing and Validating System Data]
- [Microsoft Product Integrations]
Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)

You can periodically update your application's database from the database that is used by Microsoft Dynamics GP, or you can write code that will read from the database that is used by Microsoft Dynamics GP from your own application. For example, you could query a list of items for a Web site every time the page opens, or you could write code in your application that queries the current list price for an item stored in the Microsoft Dynamics GP database.

**Table 7.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read the data for any application that needs the data and place custom requester services on all SQL tables within Microsoft Dynamics GP.</td>
<td>eConnect</td>
</tr>
<tr>
<td>This tool allows manual access, automated access, or a combination of the two to generate the data into an XML document. You also might consider using Microsoft BizTalk® Server in conjunction with eConnect to transfer data to another application.</td>
<td></td>
</tr>
<tr>
<td>Read the data for any application.</td>
<td>ODBC Drivers</td>
</tr>
<tr>
<td>The Microsoft Dynamics GP Developer Resources documentation identifies some of the tables that might be read by a developer's application.</td>
<td></td>
</tr>
<tr>
<td>Read data using native Dexterity commands, if Dexterity already is being used for other scenarios.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>Read and display data in Business Portal for any application.</td>
<td>Visual Studio .NET, Rational XDE and its code generator, the Query Web Service, result viewers, and the Business Portal SDK.</td>
</tr>
</tbody>
</table>
Subscribing to Changes within Microsoft Dynamics GP Data

You can update your application's database while the Microsoft Dynamics GP end user makes changes within Microsoft Dynamics GP. For example, the end user could add or change a customer in the Microsoft Dynamics GP Customer Maintenance window or delete an existing vendor from the Microsoft Dynamics GP Vendor Maintenance. To keep your data synchronized with the data stored in Microsoft Dynamics GP, you need to create code to update your application’s database as these changes occur in Microsoft Dynamics GP.

Table 8.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Watch for changes made to the data within Microsoft Dynamics GP for any application that needs the data.</td>
<td>eConnect</td>
</tr>
<tr>
<td>This allows the developer to watch SQL tables as they are created, modified, and deleted. An XML document is created that shows the results of the database operation.</td>
<td></td>
</tr>
<tr>
<td>Watch for changes to selected records by either watching for user activity in the windows or on the tables within Microsoft Dynamics GP.</td>
<td>Dexterity and Continuum API</td>
</tr>
<tr>
<td>Both of these tools allow a user to place watch triggers to watch for either user interface activity or database activity and to cause other code to execute when these triggers are fired.</td>
<td></td>
</tr>
</tbody>
</table>

Note  It is possible to place SQL database triggers and access data through ODBC drivers. However, this method is not recommended because you may need to make modifications for each product upgrade.

Writing Simple Data to Microsoft Dynamics GP

You can update basic setup data within Microsoft Dynamics GP.

Table 9.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write to the Microsoft Dynamics GP SQL database, if the setup data is not complex and there is no need for intense data validation.</td>
<td>ODBC drivers</td>
</tr>
<tr>
<td>Use the Microsoft Dynamics GP Developer Resources documentation as a reference for more information about various tables and fields.</td>
<td></td>
</tr>
<tr>
<td>Write to the Microsoft Dynamics GP SQL database from within a custom interface designed in Dexterity.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>For This Task</td>
<td>Use This Tool</td>
</tr>
<tr>
<td>------------------------------------------------------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Interact with a Microsoft Dynamics GP window, rather than directly with the SQL database.</td>
<td>Continuum API</td>
</tr>
</tbody>
</table>

### Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)

You can update data within Microsoft Dynamics GP and ensure that the updated data is validated by the Microsoft Dynamics GP business logic. For example, you could create new transactions within a Microsoft Dynamics GP module, such as Receivables Management, or create new master records such as Customers, Vendors, Accounts, and Employees.

**Table 10.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access the Microsoft Dynamics GP business logic through a COM or Web services interface.</td>
<td>eConnect</td>
</tr>
<tr>
<td>Define your own transactions and utilize eConnect application programming interfaces (APIs) for integration.</td>
<td>eConnect</td>
</tr>
<tr>
<td>Interact with a Microsoft Dynamics GP window directly.</td>
<td>Continuum API</td>
</tr>
<tr>
<td>Update the Microsoft Dynamics GP SQL database using an interface created in Dexterity.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>Create a new business component, query page, Web Part, or portal page for Business Portal to integrate with Microsoft Dynamics GP.</td>
<td>Visual Studio .NET, Rational XDE and its code generator, the Entity Relational Map Generator, and the Entity Association Map Generator to write the application, and then the Business Portal SDK for integration.</td>
</tr>
</tbody>
</table>

**Note** Using ODBC drivers for this task is not recommended. However, if you need to write directly to the SQL tables through ODBC drivers, information about common transaction and master records is available in the Microsoft Dynamics GP Developer Resources documentation.
Writing and Validating System Data

You can update data within Microsoft Dynamics GP at a system level that requires complex data validation. For example, you may want to access the next document number for a transaction record or update inventory allocations.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update data with complex validation if you are already using Dexterity for other tasks.</td>
<td>Dexterity</td>
</tr>
<tr>
<td>Update data with complex validation if you are already using the Continuum API for other tasks.</td>
<td>Continuum API</td>
</tr>
</tbody>
</table>

Note: It is possible to write directly to the SQL tables through ODBC drivers. This method is not recommended because you may need to make modifications for each product upgrade.

Microsoft Product Integrations

Microsoft Dynamics GP integrates with various other Microsoft products.

<table>
<thead>
<tr>
<th>Product</th>
<th>Integration Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Dynamics CRM</td>
<td>Enter sales orders in Microsoft Dynamics CRM, then track and use them in Microsoft Dynamics GP.</td>
</tr>
<tr>
<td>Microsoft Retail Management System</td>
<td>You can transfer master records and selected transaction data from Microsoft Retail Management System to Microsoft Dynamics GP. You can see Retail Management System transaction data in Microsoft Dynamics GP.</td>
</tr>
<tr>
<td>Product</td>
<td>Integration Points</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
  • Using SmartList and export solutions, you can export data to Excel or Microsoft Office Word and specify macros that will run automatically to format the data or complete calculations.  
  • Using Microsoft Office SmartTag Manager with Office and Microsoft Dynamics GP, you can access detailed information in Office about specific item, customer, vendor, and employee records stored within Microsoft Dynamics GP.  
  • Because Microsoft Dynamics GP data is stored in a SQL Server database, you can use the Data Connection wizard in Excel to access Microsoft Dynamics GP data.  
  • You can display Office Web Parts in Business Portal pages.  
  • The Excel-based Budgeting tool in Microsoft Dynamics GP allows you to export a range of accounts – for example, all the accounts in a department – from Microsoft Dynamics GP to Excel, along with historical years and the current year’s actual amounts so that each department manager can develop his or her budget. You then can import the updated Excel worksheets into Microsoft Dynamics GP.  
  • You can use the Letter Writing Assistant in Microsoft Dynamics GP to create letters in Word that you can send to customers, vendors, applicants, or employees using Microsoft Dynamics GP data and Word document templates. |
Completing a Batch Integration

Batch integrations allow you to import or export large batches of data that is collected over a period of time. Whereas with real-time integrations data is updated continually and automatically on a regular schedule, batch integrations are completed manually as they are needed. Several tools allow integrations in which data is written to and read from Microsoft Dynamics GP in batches.

The following topics are discussed:

- **Reading Existing Data from Microsoft Dynamics GP (Batch Integration)**
- **Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)**

Reading Existing Data from Microsoft Dynamics GP (Batch Integration)

You can periodically update your application's database based on information within the database for Microsoft Dynamics GP. This can be done on a schedule using a scheduling tool, or can be initiated by the user.

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Place custom requester services on all SQL tables within Microsoft Dynamics GP.</td>
<td>eConnect</td>
</tr>
<tr>
<td>This allows for manual access, automated access, or a combination of the two to generate the data as an XML document.</td>
<td></td>
</tr>
<tr>
<td>Read the data for any application that needs the data.</td>
<td>ODBC drivers</td>
</tr>
<tr>
<td>The Microsoft Dynamics GP Developer Resources documentation identifies some of the tables that might be read by a developer's application.</td>
<td></td>
</tr>
<tr>
<td>Place SQL database triggers and store the data in temporary tables or in a text file until it is ready to be processed.</td>
<td>Microsoft SQL Server</td>
</tr>
<tr>
<td>The Microsoft Dynamics GP Developer Resources documentation identifies some of the tables that might be read by a developer's application.</td>
<td></td>
</tr>
</tbody>
</table>
For This Task | Use This Tool
---|---
Read data for Dexterity-based applications. | Dexterity
Watch for batch activity at the user interface level within Microsoft Dynamics GP, such as posting or monthly reports. | Continuum API

**Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)**

A common scenario is to pass a large number of transactions and master records into Microsoft Dynamics GP. First, your application needs to maintain a list of the records to be passed. Your application then publishes the list to one of the tools in the following table, and then the tool imports the data into Microsoft Dynamics GP.

**Table 14.**

<table>
<thead>
<tr>
<th>For This Task</th>
<th>Use This Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perform batch mode integration to write validated data.</td>
<td>Integration Manager</td>
</tr>
<tr>
<td>Write validated data to Microsoft Dynamics GP in batch mode through a COM or Web services interface to the Microsoft Dynamics GP business logic.</td>
<td>eConnect</td>
</tr>
<tr>
<td>Define your own transactions and utilize eConnect application programming interfaces (APIs) for integration.</td>
<td>eConnect</td>
</tr>
<tr>
<td>Interact with a Microsoft Dynamics GP window directly and automate the window through code.</td>
<td>Continuum API</td>
</tr>
<tr>
<td>Write transactions and master records using a Dexterity based application.</td>
<td>Dexterity</td>
</tr>
</tbody>
</table>

**Note**  Using ODBC drivers for this task is not recommended. But if you need to write directly to the SQL tables through ODBC drivers, information about common transaction and master records is available in the Microsoft Dynamics GP Developer Resources documentation.

**Understanding the Tools**

Microsoft Dynamics GP tools support interconnected business solutions for the back office, front office, and e-business. Microsoft Dynamics GP allows you to write integrated applications, move data from existing sources to your new applications, and modify each customer’s application to meet specialized needs.

To validate the integrity of data from integrated applications, many of the tools can access Microsoft Dynamics GP business logic. The business logic restricts or qualifies data as it is transferred into the Microsoft Dynamics GP database. This ensures that only data meeting your criteria will be committed to the database. Customizing
business logic allows you to map and transform data, to define events triggering functionality, to execute commands, to provide feedback to the user or error log, and to accept input from the user. Developers can call existing business logic, write new business logic, or modify existing business logic. Scripting languages such as SQL, sanScript (the scripting language of Dexterity), or Microsoft VBA can be used to define the logic.

The following topics are discussed.

- eConnect
- Integration Manager
- ODBC drivers
- Modifier with VBA
- Extender
- Dexterity
- Microsoft Dynamics GP Developer Resources documentation
- Continuum API
- Business Portal SDK

**eConnect**

The eConnect Runtime together with the eConnect SDK is a set of development utilities and APIs that integrate outside data sources with Microsoft Dynamics GP components and Microsoft Dynamics GP back office document transactions.

You can obtain eConnect from the Microsoft Dynamics GP CD. If you’re an MSDN Universal subscriber, you can download the Microsoft Dynamics GP CD images from the MSDN subscriber downloads area. Updated versions of eConnect can be obtained from the CustomerSource extranet. Updated versions of the eConnect documentation can be viewed in the MSDN Library.

For more information about eConnect, see the Customization and Integration Tools User Guides for Microsoft Dynamics GP 9.0 page at the Microsoft Dynamics GP Assistance Center Web site.

**Advantages of eConnect**

The following table lists the advantages that eConnect provides when you need to connect to external data sources.

**Table 15.**

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reusable code</td>
<td>eConnect uses a series of transactional business objects that can be reused with little or no modification.</td>
</tr>
<tr>
<td>Benefit</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Standard technologies</td>
<td>eConnect uses Microsoft SQL Server technology, Visual Basic objects, stored procedures, BizTalk Server, COM, Microsoft Message Queuing, and XML. The business logic is stored as SQL stored procedures. This allows you to select your tool of choice for connectivity, so you do not need to learn new technologies.</td>
</tr>
<tr>
<td>Quicker development</td>
<td>You can focus on XML and business logic development without rebuilding transport protocols. For example, you can create an XML document for a specific transaction that utilizes a custom application in the back office. eConnect connectivity tools transfer that document from XML to the back office. ISVs using eConnect can rapidly integrate Web solutions through Web services.</td>
</tr>
</tbody>
</table>

### Skills Needed to Use eConnect

eConnect requires advanced programming skills and knowledge of Microsoft systems. You need to be familiar with the business logic implementation of your application and have experience with XML coding, XML schemas, and SQL programming and procedures. You may also need experience with one or more of the following:

- Microsoft Windows® network administration
- Microsoft Internet Information Services (IIS) 5.0 or later
- Microsoft Message Queueing (MSMQ) 2.0 or later
- Microsoft SQL Server 2000 or later
- BizTalk Server 2002 or later
- Microsoft Data Access Components (MDAC) 2.8 or later
- Database for Microsoft Business Solutions—Great Plains Release 8.0 or Microsoft Dynamics GP Release 9.0

Developers can modify eConnect or write a new version. Technical documentation is available to assist you and provide examples for writing your own integrations.

### Capabilities of eConnect

With eConnect, you can tie your systems together in real time with multiple integration points. eConnect supports integration between data from other applications and your Microsoft Dynamics GP application, enabling high volume and high speed in back office transactions. Data can be pushed into Microsoft Dynamics GP from diverse application types such as Web storefronts, Web services, legacy applications, customer relationship management applications, and point-of-sale applications.
eConnect uses Microsoft Dynamics GP business logic to ensure that data is validated and that well-formed XML documents are returned on demand. The XML documents display extensive details from back office data. You can update, insert, and delete back office transactions, and you can send back office requests to view specific documents or lists of documents.

eConnect can filter data and apply test criteria to the data before or after you import the data. You can extend the business logic to process incoming orders, invoices, or sales documents. For instance, you can limit the values that can be entered in fields, require permission for users to conduct transactions that exceed a preset monetary value, or set a default value for a field.

eConnect provides multiple methods for adding new business logic. In addition to adding script directly to the business object—as you do in Visual Basic Scripting Edition (VBScript) with tools such as Dexterity, Integration Manager, or Modifier with VBA—you can add XML nodes to the document schema in eConnect. The current nodes map directly to the SQL-based business logic on the back end, and you add the business logic in SQL to route the added XML node. The data is integrated using procedures that are stored in a Microsoft SQL Server database. Pre-SQL and post-SQL stored procedures are attached to each schema, and developers can use them to extend the integration based on their business needs.

Components of eConnect

eConnect contains an extensive set of integration components and samples. The sample integration components consist of two Visual Basic applications that demonstrate how to call the COM objects directly and how to load documents into a message queue. Developers can use these samples to implement eConnect solutions and to extend their business logic. A full SDK including additional samples is available to assist you in writing your own integrations. Other components include the following:

- A Windows Enterprise Application Integration assembly built on the Microsoft .NET Framework. This is a transport mechanism for delivering Message Queuing messages to and from the back office.
- Transaction schemas and XML examples that can be utilized with all of the eConnect connectivity tools.
- A serialization API that can dynamically create XML documents.
- Services for Microsoft Windows Server™ 2003 for transporting XML documents through Message Queuing.
- BizTalk Server adapters that can be utilized to quickly snap eConnect into your existing BizTalk Server implementation.
- COM objects or ActiveX® Controls can be utilized independently or in conjunction with other eConnect tools to access back office transactions.
There are more than 40 predefined integration areas available with eConnect. The following is a sample of back office integrations that eConnect supports:

- Shared Master (Accounts, Vendors, Customers, etc.)
- General Ledger
- Sales Order Processing
- Payables Management
- Receivables Management
- Purchase Order Processing
- Inventory Transfers
- Fixed Assets
- Project Accounting

**Related Tasks**

- Developing a New Application
- Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)
- Reading Existing Data from Microsoft Dynamics GP (Batch Integration)
- Subscribing to Changes within Microsoft Dynamics GP Data
- Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)
- Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)

**Integration Manager**

Integration Manager is an integration tool used to convert and transfer data from most database formats or desktop applications into Microsoft Dynamics GP. Integration Manager imports or exports data from a variety of sources including comma-delimited and tab-delimited text files and most ODBC driver data sources.

Integration Manager is a component-based application consisting of a core conversion engine and a set of adapters. Adapters for converting information for different formats are available for several integrating systems. The adapters process the business logic. The data source adapter processes the extraction logic, which imposes restrictions or qualifications about accepting or rejecting data. The destination adapter processes the insertion logic, which determines how to insert the new data or how to update the database.

For more information about Integration Manager, see the Customization and Integration Tools User Guides for Microsoft Dynamics GP 9.0 page at the Microsoft Dynamics GP Assistance Center Web site.
**Advantages of Integration Manager**

The following table lists the advantages that Integration Manager provides for converting and transferring data.

**Table 16.**

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database knowledge not required</td>
<td>Integration Manager assists you in mapping your data source to the Microsoft Dynamics GP objects without having to refer to the tables directly.</td>
</tr>
<tr>
<td>Option to customize</td>
<td>All integration data is processed through Microsoft Dynamics GP business objects. You can customize these business objects to ensure that data meets your needs. Integration Manager supports COM, VBScript, and SQL commands.</td>
</tr>
<tr>
<td>Minimal maintenance required for upgrades</td>
<td>After the integration is defined, the mappings and table formats have minimal changes between system upgrades. The object browser in Integration Manager simplifies ongoing maintenance.</td>
</tr>
</tbody>
</table>

**Skills Needed to Use Integration Manager**

End users and developers can use the Integration Manager graphical interface to visually map data from one format to another. Integrations can be saved and shared among other users.

Using advanced features or creating a complex integration requires development experience. A background with Microsoft Dynamics GP, database maintenance, SQL commands, and a scripting language (such as VBA or VBScript) is helpful.

**Capabilities of Integration Manager**

Integration Manager allows you to complete your own integrations, by graphically mapping fields between the source and target databases, and by selecting which rules to validate the data against. You can define SQL queries to complement your integrations. A simplified import and export feature within Integration Manager allows basic integration capabilities with limited data validation. Extensive error checking and logging allow you to review the results of any integration quickly.

In addition to simple end-user integrations, Integration Manager also supports customization through VBScript, which you can use to customize business logic to meet specific business needs. For instance, you could create a new batch name using the current system date, or create a customized document number for each transaction based on data captured through VBScript.
After you have defined the integration, you can launch it manually or automatically. From within Microsoft Dynamics GP, users explicitly launch integrations from a menu command. This allows one-time or infrequent data transfers, such as initially populating a new database. Integration Manager also can schedule integrations to run as part of a batch file or using Scheduled Tasks in Windows. This permits users to incorporate the integration as an integral part of another process or for routine transfers, such as overnight transactions or to offices in different time zones.

Like many of the Microsoft Dynamics GP custom development tools, Integration Manager is COM-compliant. Using this industry standard ensures that existing integrations will work with product upgrades. The underlying COM objects are represented as Microsoft Dynamics GP business objects. This abstraction layer allows users and developers to be one level removed from the implementation details of the database. Unlike database-level integrations, which are susceptible to changes in table structure during a product upgrade. Integration Manager business objects minimize the impact of table structure changes.

**Components of Integration Manager**

Integration Manager includes the following destination adapters:

**Table 17.**

<table>
<thead>
<tr>
<th>Adapter</th>
<th>Does This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Dynamics GP destination</td>
<td>Creates validated conversions in defined Microsoft Dynamics GP destinations.</td>
</tr>
<tr>
<td>adapter</td>
<td></td>
</tr>
<tr>
<td>SQL-Optimized destination adapter</td>
<td>Completes streamlined and faster conversions specifically for Microsoft SQL Server.</td>
</tr>
<tr>
<td>Fixed Assets Management destination</td>
<td>Integrates Fixed Assets Management data with Microsoft Dynamics GP.</td>
</tr>
<tr>
<td>adapter</td>
<td></td>
</tr>
<tr>
<td>Project Accounting destination</td>
<td>Integrates Project Accounting data with Microsoft Dynamics GP.</td>
</tr>
<tr>
<td>adapter</td>
<td></td>
</tr>
<tr>
<td>Direct to table</td>
<td>Creates integrations that aren't validated through an ODBC driver connection.</td>
</tr>
<tr>
<td>XML</td>
<td>Reads an XML document to import into Microsoft Dynamics GP. It also allows you to create XML documents from any type of data source.</td>
</tr>
</tbody>
</table>

For other data conversion options, see ODBC drivers.

**Related Tasks**

- Developing a New Application
- Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)
Open Database Connectivity (ODBC) Drivers

Open database connectivity (ODBC) drivers are integration tools to connect client applications to ODBC-compliant databases. This allows applications to share data by using a common transfer protocol.

ODBC is an industry standard for accessing databases. Database sources may be from various manufacturers and have different data formats, but as long as the database implements ODBC, information can be imported or exported by other databases or applications. Each ODBC driver is specific to a particular database management system.

Use the Microsoft Dynamics GP Developer Resources documentation as a reference for more information about the various tables and fields in the Microsoft Dynamics GP database.

Advantages of ODBC Drivers

The following table lists advantages that ODBC drivers provide as a foundation for transferring information between ODBC-compliant databases.

Table 18.

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry standard</td>
<td>Because ODBC is an industry standard for exchanging information among databases, you don't need to learn new technologies or risk being locked into proprietary systems.</td>
</tr>
<tr>
<td>Data exchange</td>
<td>Data from ODBC-compliant databases can be seamlessly imported or exported between applications. Many database manufacturers support this standard, ensuring that you can integrate with a wide range of systems.</td>
</tr>
</tbody>
</table>

Skills Required to Use ODBC Drivers

Developers who write applications that incorporate access to ODBC databases will require extensive knowledge of the database's table structure and have experience writing and maintaining databases. Developers must have an understanding of an ODBC driver interface and API functions.

Capabilities of ODBC Drivers

ODBC drivers control data import or export requests so that the information conforms to the formats and structures of the system that owns the data. Systems require one or more drivers to process ODBC requests. They are generally installed during the installation of the respective product and can be managed (for example, starting, stopping, or updating drivers) by the database or by the computer's operating system.

By default, Microsoft Dynamics GP uses the ODBC drivers installed by Microsoft SQL Server. For other databases, each manufacturer's driver capabilities might vary.
The drivers read from and write to database tables. Of the data integration options that Microsoft Dynamics GP provides, programming ODBC drivers is the most technically complex solution. Developers have direct low-level access to the database tables and must be experienced with database development.

For other data conversion options, see Integration Manager.

Related Tasks

- Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)
- Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)
- Writing Simple Data to Microsoft Dynamics GP

Modifier with Microsoft Visual Basic for Applications (VBA)

Modifier with VBA is a customization tool used to modify the appearance and functionality of any Microsoft Dynamics GP window.

For more information about Modifier with VBA, see the Customization and Integration Tools User Guides for Microsoft Dynamics GP 9.0 page at the Microsoft Dynamics GP Assistance Center Web site.

Advantages of Modifier with VBA

The following table lists the advantages that Modifier with VBA provides for modifying existing Microsoft Dynamics GP applications.

Table 19.

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple for end users</td>
<td>Modifier with VBA uses common graphics drawing techniques. These include using a tool palette and double-clicking an object to see its specific information.</td>
</tr>
<tr>
<td>Safe</td>
<td>Original information cannot be lost or overridden. The modified files are stored separately from the original files.</td>
</tr>
<tr>
<td>VBA and COM</td>
<td>Modifier with VBA uses VBA and COM technologies. This extends the ability to customize, share data, and access functionality in other applications.</td>
</tr>
</tbody>
</table>

Skills Needed to Use Modifier with VBA

Modifier with VBA can be used by end users and developers. End users can use the graphical interface to make simple or cosmetic changes to the system. Experienced users can add or delete items such as windows, controls, or fields. No developer experience is required, although knowledge of Microsoft Dynamics GP windows is helpful. Customizations can be saved and shared among other users.
Using advanced features or creating a complex integration requires development experience. A background with Microsoft Dynamics GP, database maintenance, SQL commands, and a scripting language (such as VBA) is helpful.

Capabilities of Modifier with VBA

Modifier with VBA uses embedded VBA. Developers can add scripts, tools, and access to Microsoft Office objects to incorporate new levels of functionality into their applications. Modifier with VBA allows you to alter the appearance of the Microsoft Dynamics GP application in multiple ways.

- Cosmetic changes can be made to individual windows, including the color, font, text, and graphics displayed within a window. In addition, text can be changed. For instance, you could change the name of a menu item from "Financial" to "General Ledger."

- Information displayed within a window can be altered, moved, set to a default, deleted, or protected. This allows important information to be moved to a prominent position, and less important information to be deemphasized or removed entirely. For instance, a technical support user may not need to see the customer's point of purchase location that a sales support user requires. In this case, you can remove the point of purchase information from only the technical support team's window. In a similar fashion, that point of purchase information could still be displayed but changed to a read-only field, because the technical support team may need to know that information but should not be able to change it.

- New information or business logic can be added as the project evolves. This can include adding new fields or windows rather than just rearranging existing ones. For instance, you may need to limit the values that can be entered in fields or require permission for users to conduct transactions that exceed a preset monetary value.

Modifier with VBA is COM-compliant. Using this industry standard ensures that your applications can communicate with other applications such as Microsoft Office, other COM-compliant applications, or ActiveX objects. Common examples of ActiveX objects that you might want to integrate include bar code printers, credit card authorization controls, package tracking and cost calculators, and postal code databases. They may be developed in-house or purchased through third parties.

To ensure data integrity, changes made with Modifier with VBA are saved as a separate file, apart from the system resources. In this way, the original information is protected from accidental modification. By providing separate files, individual users or groups of users can receive custom changes without affecting other users, compromising the underlying business logic, or endangering the accounting integrity.

Modifier with VBA does not create new applications as Visual Basic and C++ do. Instead, Modifier with VBA is designed to alter existing applications. Although it is a powerful tool for making modifications (to the point of adding new windows and fields), it is not recommended for creating new applications or for making changes that significantly alter the nature of the application. For example, do not use Modifier with VBA to transform an existing sales application into an accounts receivable application. It is better to design a new application. To create a new application, to modify a commercially available application for multiple locations, or to create a mass-marketed application, see Continuum API or Dexterity.
Related Tasks

- Rearranging Fields
- Adding Fields
- Changing the Properties of a Field
- Changing Text Labels
- Customizing the Business Logic

Extender

Extender is a customization and integration tool that can be used to modify Microsoft Dynamics GP applications and import data into Microsoft Dynamics GP from external sources.

For more information about Extender, see the Microsoft Dynamics GP 9.0 User Guides page at the Microsoft Dynamics GP Assistance Center Web site.

Advantages of Extender

The following table lists the advantages that Extender provides for modifying existing Microsoft Dynamics GP applications.

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software development experience not required</td>
<td>Use a graphical user interface to customize Microsoft Dynamics GP.</td>
</tr>
<tr>
<td>Re-usable customizations</td>
<td>Export Extender resources in XML format so that you can import them for use with other companies within Microsoft Dynamics GP or with other Microsoft Dynamics GP installations.</td>
</tr>
</tbody>
</table>

Skills Needed to Use Extender

Extender has a graphical user interface that can be used to customize Microsoft Dynamics GP. No software development experience is required, although knowledge of Microsoft Dynamics GP is helpful. Customizations can be saved and shared among other users. Development experience is required to use Extender to create complex integrations.
Capabilities of Extender

Extender can be used to customize data collection in Microsoft Dynamics GP in multiple ways.

- New data entry windows can be added to a Microsoft Dynamics GP form. You can:
  - Link the new window to an existing window within the form, then select the type of data from the existing window to be used as the primary key for the database table that will be created to support the new window.
  - Add conditional and calculated fields to the new window and indicate whether they are required.
  - Add lookups to the window that can be linked to any Microsoft Dynamics GP table.
  - Assign shortcut keys that can be used to open the window.
  - Create SmartList favorites based on the fields and windows that you’ve created.
  - Export Extender resources in XML format so that you can import them for use with other companies within Microsoft Dynamics GP or with other Microsoft Dynamics GP installations.

- Note entry windows can be added to a Microsoft Dynamics GP form so that users can attach multiple notes to records. Notes can be automatically created based on rules. Notes can be imported from Microsoft Office Outlook 2003.

Extender does not create new applications as Visual Basic and C++ do. Instead, Extender is designed to alter existing applications. Although it is a powerful tool for making modifications (to the point of adding new windows and fields), it is not recommended for creating new applications or for making changes that significantly alter the nature of the application. For example, do not use Extender to transform an existing sales application into an accounts receivable application. It is better to design a new application. To create a new application, to modify a commercially available application for multiple locations, or to create a mass-marketed application, see Continuum API or Dexterity.

Related Tasks

- Adding Fields

Dexterity

Dexterity is a development tool used to create or modify large-volume, transaction-based client/server applications that seamlessly integrate with Microsoft Dynamics GP modules.

The Microsoft Dynamics GP and Small Business Financials Integration Guide provides information about developing Dexterity applications that integrate with Microsoft Dynamics GP. The Guide is included with Dexterity.
The Microsoft Dynamics GP Developer Resources documentation also includes information about the Dexterity-based procedures and functions that you can use to leverage the Microsoft Dynamics GP business logic.

For more information about Dexterity, see the Customization and Integration Tools User Guides for Microsoft Dynamics GP 9.0 page at the Microsoft Dynamics GP Assistance Center Web site.

**Advantages of Dexterity**

The following table lists the advantages that Dexterity provides for custom application development.

**Table 21.**

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common development environment</td>
<td>Dexterity is the same language used to develop Microsoft Dynamics GP. It consists of a full development environment created with C++, which allows you to integrate your custom modules with the standard Microsoft Dynamics GP modules.</td>
</tr>
<tr>
<td>Extensible modules</td>
<td>With Dexterity, you can extend existing Microsoft Dynamics GP modules. Dexterity implements changes by storing modifications in files separate from the system code. Individual users or user groups can change the application to suit their needs without interfering with other groups' customization requirements.</td>
</tr>
<tr>
<td>Easy creation of new vertical applications</td>
<td>You can use Dexterity to create new forms and tables and easily integrate them into the Microsoft Dynamics GP user interface.</td>
</tr>
</tbody>
</table>

**Skills Needed to Use Dexterity**

Extensive experience with software development is recommended. Dexterity is a complete development environment, and training is recommended. The following skills assist with development:

- An understanding of an application development environment such as Visual Basic or Microsoft Visual Studio.
- Knowledge of third-generation programming languages, such as C/C++ or Pascal.
- Experience using a scripting language such as VBScript. Customizations made with Dexterity require the use of sanScript, a plain English scripting language.
- An understanding of network environments and client/server architecture.
The Microsoft Dynamics GP Developer Resources documentation includes documentation and details about the integration points.

**Capabilities of Dexterity**

Dexterity is a development environment that provides you with the same complete set of components that were used to create Microsoft Dynamics GP. It allows you access to all of the Microsoft Dynamics GP resources such as windows, scripts, tables, and reports. In addition to Dexterity's native customization capabilities, you have access to the capabilities of **Modifier with VBA** when you use Dexterity. These features combined include graphical interface technologies, multiple-document interface (MDI) support, platform-native controls, background processing, and inter-application communications capabilities.

The programming environment includes a graphical editor to construct and arrange system components such as windows and controls. If you have used the graphical editor in Visual Basic or Visual Studio, you will be comfortable with the Dexterity programming environment. Development components include a forms builder, a database manager, an embedded macro system, and a scripting language and debugger. Dexterity supports embedded technologies such as COM and VBA.

The Dexterity scripting language is named sanScript and is the basis for all business logic in Microsoft Dynamics GP. The scripted procedures not only define actions but also define triggers that initiate actions. The wide range of triggers include database events, focus events (such as entering or leaving a field), and opening or restarting a form.

Dexterity achieves a high level of customization through the use of dictionary files. The dictionary files store the compiled windows code, control code, and the sanScript scripts. These files can be distributed to your system's users. The Microsoft Dynamics GP system can load and run a virtually unlimited number of these dictionary files, and the system will run all of the appropriate procedures when an event is triggered. This multi-dictionary environment allows applications to be easily deployed to multiple sites or multiple customers. Using dictionary files protects the integrity of Microsoft Dynamics GP. Systems can have dictionary files added or removed to provide the needed customization, but the original Microsoft Dynamics GP source is never modified.

Dexterity can be used to create new or large applications, or to significantly modify an existing application. Dexterity provides a powerful foundation for large applications. It has direct access to call (but not view) Microsoft Dynamics GP procedures and functions, has access to use all predefined tables and fields, and you can use it to set up triggers for new business logic or for information captures. For these reasons, Dexterity allows the most extensive changes among the various Microsoft Dynamics GP development tools.
Related Tasks

- Developing a New Application
- Rearranging Fields
- Adding Fields
- Changing the Properties of a Field
- Changing Text Labels
- Customizing the Business Logic
- Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)
- Reading Existing Data from Microsoft Dynamics GP (Batch Integration)
- Subscribing to Changes within Microsoft Dynamics GP Data
- Writing Simple Data to Microsoft Dynamics GP
- Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)
- Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)
- Writing and Validating System Data

Microsoft Dynamics GP Developer Resources documentation

The Microsoft Dynamics GP Developer Resources documentation, which includes the Microsoft Dynamics GP SDK, provides development documentation and detailed technical information about the Microsoft Dynamics GP data formats. The resources in the documentation help you to produce products that integrate with Microsoft Dynamics GP and to customize Microsoft Dynamics GP components.

Advantages of the Microsoft Dynamics GP Developer Resources documentation

The following table lists the advantages that the Microsoft Dynamics GP Developer Resources documentation provides for custom application development.

Table 22.

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consolidated source of information</td>
<td>The information required to integrate with the Microsoft Dynamics GP data model is available in the Microsoft Dynamics GP Developer Resources documentation.</td>
</tr>
<tr>
<td>Documented data model changes</td>
<td>The Microsoft Dynamics GP Developer Resources documentation provides detailed information about database and procedural changes between product releases.</td>
</tr>
</tbody>
</table>
Skills Needed to Use the Microsoft Dynamics GP Developer Resources documentation

The information in the Microsoft Dynamics GP Developer Resources documentation is intended to assist developers who are creating applications that integrate with Microsoft Dynamics GP or who are customizing implementations of Microsoft Dynamics GP. The information is technically detailed, and developers require extensive software development experience, database knowledge, and familiarity with their product’s systems and business requirements.

Capabilities of the Microsoft Dynamics GP Developer Resources documentation

The Microsoft Dynamics GP Developer Resources documentation describes tables, data modeling, data formats, the creation of master and transaction records, and transaction posting flows. It is useful to developers who use customization and integration tools such as Modifier with VBA, the Continuum API, ODBC drivers, and Dexterity. Users who are already familiar with the Microsoft Dynamics GP development environment will find topics that describe changes from previous versions, including changes to the data model and to the Microsoft Dynamics GP dictionary.

Some components will be useful only to Dexterity or Continuum API developers. These topics include documentation that describes changes in global procedures and functions, parameters for all procedures and functions, and changes to forms from various released versions. In those cases, that information is explicitly identified by notes such as "These topics are intended only for Dexterity developers."

Components of the Microsoft Dynamics GP Developer Resources documentation

The Microsoft Dynamics GP Developer Resources documentation includes sections describing the following:

Table 23.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Details Documented</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Dynamics GP integration</td>
<td>Procedures and functions, table integration, warning messages, and dictionary changes.</td>
</tr>
<tr>
<td>ERwin models, a graphical model of the database</td>
<td>Provides ERwin models for the Microsoft Dynamics GP modules.</td>
</tr>
<tr>
<td>Microsoft Dynamics GP changes</td>
<td>Includes documentation describing database and procedural changes within Microsoft Dynamics GP.</td>
</tr>
</tbody>
</table>

Related Tasks

- Developing a New Application
- Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)
Continuum Application Programming Interface (API)

Continuum is the COM API that is available for Microsoft Dynamics GP. Tools that support COM automation, such as Microsoft Visual Basic for Applications, can use the Continuum API to interact with Microsoft Dynamics GP.

For more information about the Continuum API, see the Customization and Integration Tools User Guides for Microsoft Dynamics GP 9.0 page at the Microsoft Dynamics GP Assistance Center Web site.

Advantages of the Continuum API

The following table lists the advantages that the Continuum API provides for custom application development.

**Table 24.**

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Widely-used standard Microsoft technology</td>
<td>COM is an industry-standard technology. The Continuum API is a standard COM interface for the Microsoft Dynamics GP application.</td>
</tr>
<tr>
<td>Extended integration</td>
<td>If you’re a developer creating or modifying an application using a COM-compliant language, you can write your application so that it can interact with Microsoft Dynamics GP using the Continuum API.</td>
</tr>
</tbody>
</table>

Skills Needed to Use the Continuum API

Intermediate to advanced experience with software development is recommended. The following skills assist with development:

- An understanding of the COM-capable application development environment that you will be using, such as Visual Basic.
- An understanding of the Microsoft Dynamics GP application and database structure.
- Experience with sanScript, the scripting language used to develop Microsoft Dynamics GP.

A supplement that describes the sanScript language is included with the Continuum API documentation. Complete documentation and samples for sanScript are included with the documentation that ships with Dexterity. The Microsoft Dynamics GP Developer Resources documentation includes information about Microsoft Dynamics GP integration points.
Capabilities of the Continuum API

The Continuum API allows any applications written in a COM-compliant development environment to interact with the Microsoft Dynamics GP application. The Continuum API allows three types of integration: interface, database, and process.

- **Interface integrations**: The integrating application interacts with or manipulates the user interface in Microsoft Dynamics GP. Interface integrations are the simplest integrations to create.

- **Database integrations**: The integrating application reads from or writes to the Microsoft Dynamics GP database, and can monitor database changes triggered by Microsoft Dynamics GP. A thorough understanding of the Microsoft Dynamics GP database is necessary to create database integrations. The Continuum API provides limited support for database integrations, therefore other integration methods might be more appropriate.

- **Process integrations**: The integrating application reacts to events from the Microsoft Dynamics GP system, such as posting. Process integrations require a thorough understanding of the Microsoft Dynamics GP application process that you are integrating with.

If you must make only minor changes to the Microsoft Dynamics GP application, consider using [Modifier with VBA](#) instead of the Continuum API. To create new or large applications or to significantly modify the Microsoft Dynamics GP application, use [Dexterity](#).

Components of the Continuum API

The Continuum API in Microsoft Dynamics GP can be used with any COM-compliant development environment such as Visual Basic .NET. The Continuum API Guide provides detailed information about the API. Sample applications demonstrate how the Continuum API can be used for integrations.
Related Tasks

- Developing a New Application
- Changing the Properties of a Field
- Customizing the Business Logic
- Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)
- Reading Existing Data from Microsoft Dynamics GP (Batch Integration)
- Subscribing to Changes within Microsoft Dynamics GP Data
- Writing Simple Data to Microsoft Dynamics GP
- Writing Transactions and Master Records to Microsoft Dynamics GP (Real-Time Integration)
- Writing Transactions and Master Records to Microsoft Dynamics GP (Batch Integration)
- Writing and Validating System Data

Business Portal SDK

The Business Portal SDK provides development documentation and detailed technical information about Microsoft Dynamics GP Business Portal business components, query pages, Web Parts, and portal pages. The resources in the SDK help you to produce products that integrate with Business Portal and to customize Business Portal components. See the Business Portal SDK for more information.

For more information about developing for Business Portal, see the Business Portal Developer Documentation page at the Microsoft Dynamics GP Assistance Center Web site.

Advantages of the Business Portal SDK

The Business Portal SDK includes the information required to integrate with Business Portal business components, query pages, Web Parts, and portal pages is available in the Business Portal SDK.

Skills Needed to Use the Business Portal SDK

The Business Portal SDK describes creating and using business entities, entity maps, entity associations, the Query Web Service, Web Parts, portal pages, and using security. The information in the Business Portal SDK is intended to assist developers who are creating applications that integrate with Business Portal or who are customizing implementations of Business Portal. The information is technically detailed, and developers require extensive software development experience, database knowledge, and familiarity with their product's systems and business requirements.
Capabilities of the Business Portal SDK

The SDK is for developers who use Microsoft Visual Studio .NET, the C# programming language, Rational XDE and its code generator, the Entity Relational Map Generator, and the Entity Association Map Generator for customizations and integrations. Users who are already familiar with the Business Portal development environment will find topics that describe changes from previous versions.

Components of the Business Portal SDK

The Business Portal SDK includes an Integration Guide, sample code, and information about the Microsoft Dynamics GP object model.

The Integration Guide includes the following information.

- Introduces the Microsoft Business Framework
- Describes how to create entities within business components
- Describes how to create and use entity maps
- Provides an overview of several Web Parts that can be created and displayed inside Business Portal
- Describes how Web Parts can communicate with one another
- Provides an overview of how the Query Web Service is used in Business Portal
- Provides an overview of the various types of portal pages that you can create
- Describes how security features should be implemented
- Explains packaging Business Portal integrations for distribution so that they can be installed in other Business Portal installations
- Explains how to provide online help support for your integration

Related Tasks

- [Developing a New Application](#)
- [Reading Existing Data from Microsoft Dynamics GP (Real-Time Integration)](#)
Reporting Tools

A broad range of flexible, customizable reporting tools are available for Microsoft Dynamics GP.

- Report Writer
- Microsoft Dynamics GP Analysis Cubes for Excel
- Microsoft Dynamics GP Analytical Accounting
- Microsoft Enterprise Reporting
- Crystal Reports Professional
- Microsoft FRx® Professional
- Microsoft SQL Reporting Services for Microsoft Dynamics GP
- SmartList
- SmartList Builder

See Microsoft Dynamics GP for analytics for more information.