# Message Agent

## Advanced Design and Setup Guide

Perceptive Content Version: 7.0

Written by: Product Knowledge, R&D
Date: October 2014

perceptivesoftware
from Lexmark

# Table of Contents

# Introduction

Message Agent provides robust, server-side integration opportunities with Perceptive Content by using standards-based SOAP XML web service calls. You can use Message Agent to support application integrations in all market segments. Based on standard protocols for web services, specifically SOAP and WSDL, Message Agent provides customers with maximum flexibility to incorporate Perceptive Content functionality directly into their products and enterprise systems.

Following are some examples of how Message Agent can work for you:

- Update a branch office loan application immediately as mortgage documents are received and scanned at the home office.
- File insurance claims directly into Perceptive Content from a PDA or Tablet PC and route to the appropriate workflow queue.
- Analyze and act on workflow items from your My Workflow portal in the corporate portal.
- Click “Search” in the My Documents area of the Physician’s Portal and get a list of pending reports – select the result and view in WebNow.
- Notify students on their mobile device when their financial aid package is approved.

The power of Message Agent is that it lets you interact seamlessly with Perceptive Content Server to increase efficiencies by using your existing tools and without exposing end-users to new technology.

# A standards-based approach

Message Agent relies on the HTTP/HTTPs transport protocol, and Extensible Markup Language (XML) rules for exchanging structured data using standard Internet protocols.

Other supported standards include:

- **Simple Object Access Protocol (SOAP) 1.1 and later.** The preferred protocol for XML message exchange over HTTP, SOAP describes the contents of a message and how to process it.
- **Web Services Description Language (WSDL) 1.1 and later.** WSDL describes in XML how to access and act on each web service operation.
- **Universal Description, Discovery, and Integration (UDDI).** A set of protocols that provide a listing or directory of services available in a network.

Message Agent provides direct support for SOAP and WSDL, and indirect support for UDDI. Customers can list Message Agent services in a UDDI directory.

The benefit of using protocols that are language and platform independent is application interoperability, which means you can programmatically exchange data between otherwise independent products and systems.

With Message Agent, you can continue to use your preferred tools and programming language. The only requirement is that the language provides a WSDL 1.1 compliant code generator.

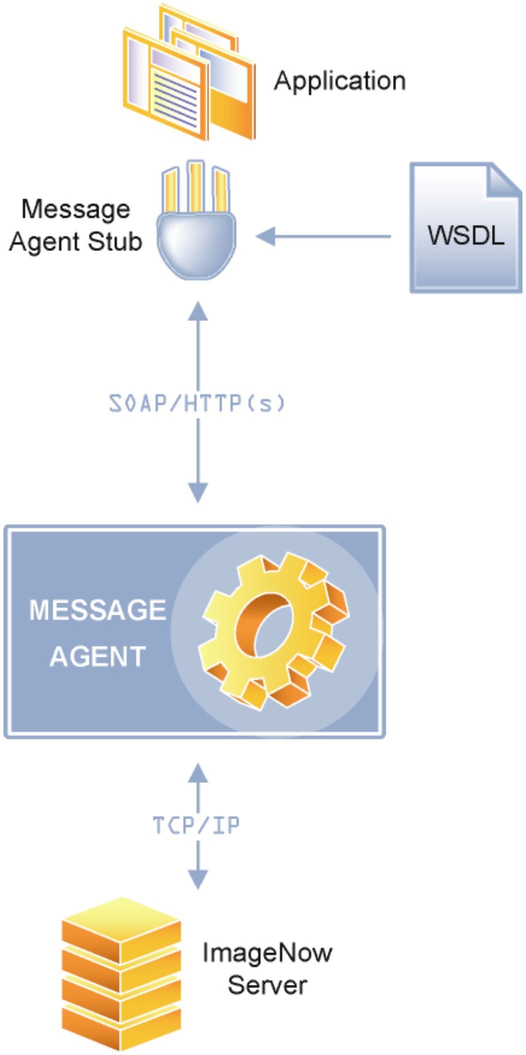
# A solid architecture

Message Agent exposes Perceptive Content Server functionality that the authorized client business application can then access.

The figure below shows how a client program makes a call to Message Agent using SOAP-XML over HTTP/HTTPs while the Message Agent server itself uses TCP/IP to communicate with the Perceptive Content Server.

By using TCP/IP, Message Agent can install anywhere on the network, including the Perceptive Content server if desired, giving customers maximum flexibility when designing for scalability and performance.

The WSDL files and SOAP can work with applications in any programming language and platform. You create the client proxy code stubs that you integrate into your client application. Moreover, with the details of deconstructing SOAP messages handled by Message Agent, developers can focus on writing the client-side business logic.



## Design and performance

Message Agent is designed to be installed on a server, and can communicate over the network with Perceptive Content software and third-party software.

The architecture supports asynchronous, synchronous, and notification communication patterns by using standard XML-SOAP message formats.

Message Agent is multi-threaded allowing multiple client requests to be executed concurrently. For secure client to server communication, Message Agent supports SSL.

## Key Design Objectives

- **Loosely-coupled server-side integration**

  Message Agent is not bound tightly to other applications to provide application interoperability. Unlike product designs that require tight integration to communicate, web services are loosely coupled. The benefit here is that the independence of each service makes the design flexible and easy to modify without affecting unrelated areas in the application.

  Consider the difference between a telephone call and an e-mail. The telephone call is tightly bound, or synchronous, requiring both parties to participate in order to make the communication transaction; the e-mail is loosely coupled, or asynchronous, because it does not require both parties to be online simultaneously to communicate. Message Agent, like the e-mail server, provides the loose coupling, and improved business agility is a direct result of a loosely-coupled design: You can choose the services you want when you want.

- **Universal API and standards support**

  A universal API lets Message Agent support any programming language. Using relevant standards also keeps Message Agent competitive and market-oriented.

  XML messaging that follows web service standards encourages customer product adoption and gives customers access to the many free web services, which in turn accelerate project production while reducing costs. With a common vocabulary and language chosen – XML, WSDL, SOAP – it is much less complicated to hook applications together. Web standards help customers to leverage their existing investments.

- **Keep design simple and provide immediate value**

  A challenge to designing any integration layer is the ability to setup and install quickly. Message Agent can deploy quickly, and robust code examples are provided to assist you during the integration project. Perceptive Software, has invested in third-party technology rather than free, unsupported open-source software.

  By using a third-party technology and providing dedicated customer support, Message Agent is optimized to deliver performance, scalability, and features beyond what open-source functionality can provide. The benefits of meeting all of these key design objectives for Message Agent are summarized below.

**Table 1:** Features and benefits of Message Agent.

| Feature | Benefit |
|---|---|
| Perceptive Content functionality implemented as web services over firewall friendly HTTP or HTTPS | ***Streamline business processes***<br>- Notify applications of new documents for timely customer response<br>- Consolidate documents into one centrally managed location<br>- Place documents into workflow from any location<br>- Manage documents in workflow without leaving host application<br>- Monitor status of workflow processes from any application |
| Standards support<br>(WSDL, SOAP, XML) | ***Accelerate time to market & leverage existing investments***<br>- Develop in language of choice (C++, C#, Java, VB, JavaScript, Perl)<br>- Many free toolkits & IDE plug-ins<br>- Rapid development & auto-code generation<br>***Ensure compliance***<br>- Wide interoperability and cross platform support<br>***Prevents vendor lock-in***<br>- Use smaller less expensive systems from many vendors |
| Asynchronous Messaging \| Loosely-couple systems together | ***Increase business agility to meet changing requirements***<br>- Reduce complexity of integrations<br>- Easily share services<br>- Support long-running real-world business processes |

# Available web services

You can use these available operations in each web service to write your client business application.

| Endpoint | Operation / Web Service |
|---|---|
| Access<br>General Authentication Web Services | ACCESS_SESSION_BEGIN_USING_PASSWORD<br>ACCESS_SESSION_END<br>ACCESS_USER_CHECK |
| Document Web Services | CAPTURE_PROFILE_CAPTURE_DOC_OBJECT<br>CAPTURE_PROFILE_GET<br>CAPTURE_PROFILE_GET_INFO_LIST<br>CAPTURE_PROFILE_END_UPLOAD<br>CAPTURE_PROFILE_START_UPLOAD<br>CUSTOM_PROPERTY_GET<br>CUSTOM_PROPERTY_GET_ALL<br>DIGITAL_SIGNATURE_REASON_GET_LIST<br>DOC_SET_NAME<br>DOCUMENT_ADD_TO_PROJECT<br>DOCUMENT_CHECK_IN<br>DOCUMENT_CHECK_IN_FILE<br>DOCUMENT_CHECK_OUT<br>DOCUMENT_CHECK_OUT_FILE<br>DOCUMENT_COPY<br>DOCUMENT_DELETE<br>DOCUMENT_DRAWER_GET_LIST<br>DOCUMENT_GET_VERSION_LIST<br>DOCUMENT_GET_VERSION_HISTORY<br>DOCUMENT_INFO_GET<br>DOCUMENT_KEYWORDS_GET<br>DOCUMENT_KEYWORDS_SET<br>DOCUMENT_KEYWORDS_DELETE<br>DOCUMENT_LOCK<br>DOCUMENT_MOVE<br>DOCUMENT_NOTIFY_NEW<br>DOCUMENT_RELATED_WORKSHEET_RESPONSE<br>DOCUMENT_REMOVE_FROM_PROJECT |

| Endpoint | Operation / Web Service |
|---|---|
| | DOCUMENT_RETURN<br>DOCUMENT_RETURN_SWA<br>DOCUMENT_RETURN_SEND_TO_USER<br>DOCUMENT_SEARCH_QUERY<br>DOCUMENT_SET_VERSION_COMMENTS<br>DOCUMENT_SET_VERSION_PUBLIC<br>DOCUMENT_SET_VERSION_PRIVATE<br>DOCUMENT_SIGN<br>DOCUMENT_SIGNATURE_VERIFY<br>DOCUMENT_SIGNATURE_VERIFY_ALL<br>DOCUMENT_SIGNATURE_GET_LIST<br>DOCUMENT_SIGNATURE_VOID<br>DOCUMENT_STORE<br>DOCUMENT_STORE_SWA<br>DOCUMENT_SUBMIT_TO_CONTENT<br>DOCUMENT_TYPE_GET<br>DOCUMENT_TYPE_GET_ALL<br>DOCUMENT_TYPE_GROUP_GET<br>DOCUMENT_TYPE_GROUP_GET_ALL<br>DOCUMENT_TYPE_GROUP_UPDATE<br>DOCUMENT_VERSION_CONTROL_PROMOTE<br>DOCUMENT_VERSION_CONTROL_ADD<br>DOCUMENT_VERSION_CONTROL_REMOVE<br>DOCUMENT_UNDO_CHECK_OUT<br>DOCUMENT_UNLOCK_ALL<br>DOCUMENT_UNLOCK<br>DOCUMENT_UPDATE_CUSTOM_PROPERTIES<br>GET_UNIQUE_ID<br>INOW_USER_GET_GROUPS<br>INOW_USER_GET_INFO<br>IN_VIEW_GET<br>IN_VIEW_RUN<br>PROJECT_CREATE<br>PROJECT_UPDATE<br>PROJECT_DELETE |

| Endpoint | Operation / Web Service |
|---|---|
| | PROJECT_GET<br>PROJECT_GET_ALL<br>PROJECT_GET_DOCUMENTS<br>PROJECT_TYPE_GET<br>PROJECT_TYPE_GET_ALL<br>SEND_EXTERNAL_MESSAGE |
| Folder Web Services | DRAWER_WITH_CREATE_PRIV_GET_LIST<br>DOCUMENT_ADD_TO_FOLDER<br>DOCUMENT_REMOVE_FROM_FOLDER<br>FOLDER_CREATE_SHORTCUTS<br>FOLDER_DELETE_SHORTCUTS<br>FOLDER_GET_DOCUMENT_ID_BY_PATH<br>FOLDER_GET_FOLDER_ID_BY_PATH<br>FOLDER_GET_OBJECT_BY_PATH<br>FOLDER_GET_PATH_FOR_OBJECT<br>FOLDER_MOVE_OBJECT_BY_PATH<br>IN_VIEW_GET_ALL |
| Form Web Services | FORM_ADD<br>FORM_ADD_DATA_DEFINITION<br>FORM_ADD_PRESENTATION<br>FORM_ADD_SHARED<br>FORM_GET<br>FORM_GET_ALL<br>FORM_GET_ALL_DATA_DEFINITION<br>FORM_GET_ALL_PRESENTATION<br>FORM_GET_ALL_SHARED_FILE<br>FORM_GET_DATA_DEFINITION<br>FORM_GET_PRESENTATION<br>FORM_GET_SHARED_FILE<br>FORM_UPDATE<br>FORM_UPDATE_PRESENTATION<br>FORM_PUBLISH_BY_FILE<br>FORM_PUBLISH_BY_ID |

| Endpoint | Operation / Web Service |
|---|---|
| Task Web Services | MY_ASSIGNED_TASKS<br>TASK_ADD_COMMENT<br>TASK_COMPLETE<br>TASK_CREATE<br>TASK_GET_ALL_REASONS<br>TASK_GET_PROPERTIES |
| Workflow Web Services | WORKFLOW_DOC_GET_INFO<br>WORKFLOW_GET_USER_DOC_INFO<br>WORKFLOW_PROCESS_GET_LIST<br>WORKFLOW_Q_GET_HOLD_REASON_LIST<br>WORKFLOW_Q_GET_LIST<br>WORKFLOW_Q_GET_ROUTES_FORWARD<br>WORKFLOW_Q_GET_USERS<br>WORKFLOW_QITEM_DELETE<br>WORKFLOW_QITEM_GET<br>WORKFLOW_QITEM_GET_HISTORY<br>WORKFLOW_QITEM_GET_INFO<br>WORKFLOW_QITEM_GET_LIST<br>WORKFLOW_QITEM_GET_NEXT<br>WORKFLOW_QITEM_GET_STATUS<br>WORKFLOW_QITEM_NOTIFY_NEW<br>WORKFLOW_QITEM_ROUTE_AUTO<br>WORKFLOW_QITEM_ROUTE_BACK<br>WORKFLOW_QITEM_ROUTE_MANUAL<br>WORKFLOW_QITEM_SET<br>WORKFLOW_QITEM_SET_HOLD<br>WORKFLOW_QITEM_SET_STATUS |

# Index