[MS-IPHWS]:
InfoPath HWS Feature Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.

- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: http://www.microsoft.com/interop/osp) or the Community Promise (available here: http://www.microsoft.com/interop/cp/default.mspx). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.

- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability
06/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
07/13/2009	1.02	Major	Changes made for template compliance
08/28/2009	1.03	Editorial	Revised and edited the technical content
11/06/2009	1.04	Editorial	Revised and edited the technical content
02/19/2010	2.0	Minor	Updated the technical content
03/31/2010	2.01	Editorial	Revised and edited the technical content
04/30/2010	2.02	Editorial	Revised and edited the technical content
06/07/2010	2.03	Editorial	Revised and edited the technical content
06/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	2.05	Minor	Clarified the meaning of the technical content.
09/27/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.

Release: Sunday, December 19, 2010

Table of Contents

1 Introduction

The InfoPath Human Workflow Services (HWS) Feature Protocol provides an interface for a protocol client to interact with a protocol server that is managing **workflow** processes which require human interaction.

A human workflow service enables the constructing, modifying, participating in, and tracking of the progress of workflows from the protocol client.

Information workers can:

- Perform workflow activities either one by one as needed or together as part of a predefined model.

- Monitor what workflow participants do as part of a day-to-day workflow.

- Track the status of workflow activities with visibility into the state of the business process.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

> **Coordinated Universal Time (UTC)**
> **globally unique identifier (GUID)**
> **GUIDString**
> **Hypertext Transfer Protocol (HTTP)**
> **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**
> **NULL GUID**
> **Unicode**
> **XML**

The following terms are defined in [MS-OFCGLOS]:

> **action**
> **activation**
> **activity**
> **participant**
> **site**
> **SOAP (Simple Object Access Protocol)**
> **SOAP action**
> **SOAP body**
> **SOAP fault**
> **SOAP fault code**
> **SOAP message**
> **target**
> **task**
> **TCP/IP**
> **trusted authentication**
> **URL (Uniform Resource Locator)**
> **Web service**
> **workflow**
> **XML namespace**
> **XML Schema**
> **XPath expression**

The following terms are specific to this document:

action instance: The runtime instance of a specific action. Action instances are building blocks for an activity flow. Several action instances can be chained together to form an activity flow, and multiple action instances of the same action can exist in a single activity flow.

Activate: The process of creating a new action instance of an action.

activation block: A grouping of steps or actions that are within an activity model and are always activated together.

activation message: An XML document that is sent to the workflow processing engine and causes a new action instance to be created.

activity flow: A running instance of a workflow that consists of a sequence of action instances and/or activity model instances. Action instances and activity model instances can be sequenced in any order to create a single activity flow.

activity model: A predefined sequence of actions.

actor: A person or process that starts or participates in an activity flow. An actor can be an initiator or a target.

ad hoc composition: A type of composition that extends an activity flow by one action instance at a time, instead of adding several action instances together in a predefined sequence.

constraint: A clause that specifies who can be the initiator, the target, or the enacted-on user of an action. Constraints define the security policy for an action.

dependent composition: An activity flow composition where the activation of one action is dependent upon the completion of another action.

finish message: An XML document that is sent to an action instance and causes the action instance to finish execution.

initiator: An actor who starts an action instance.

interrupt message: An XML document that is sent to an action instance to stop the instance.

resource property: A property that is defined by an application implementer to store additional data that is needed by the application.

step: A single action in an activity model.

task message: An XML document that is sent through a workflow system and conforms to the task message schema. An actor can communicate with an action through a task message.

task response: A response to a task from a target. A task can have zero or more responses.

task response message: An XML document that specifies the properties of a task response.

tracking system: A software component that records messages that are sent and received by workflows in a workflow system, and stores them as structured data in a data store. A tracking system enables protocol clients to query information about objects in a workflow.

workflow system: A set of software components that manages the creation and execution of a workflow.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.ietf.org/rfc/rfc2119.txt

[RFC2396] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998, http://www.ietf.org/rfc/rfc2396.txt

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, http://www.ietf.org/rfc/rfc2616.txt

[RFC2781] Hoffman, P., and Yergeau, F., "UTF-16, an encoding of ISO 10646", RFC 2781, February 2000, http://www.ietf.org/rfc/rfc2781.txt

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, http://www.ietf.org/rfc/rfc2818.txt

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, http://www.w3.org/TR/2001/NOTE-wsdl-20010315

[XMLNS] World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation 8 December 2009, http://www.w3.org/TR/REC-xml-names/

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary", March 2007.

[MS-OFCGLOS] Microsoft Corporation, "Microsoft Office Master Glossary", June 2008.

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, http://www.w3.org/TR/2003/REC-soap12-part1-20030624

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, http://www.w3.org/TR/2003/REC-soap12-part2-20030624

1.3 Protocol Overview (Synopsis)

This document specifies a protocol that enables a protocol client to manage and interact with workflows which are defined on a protocol server providing workflow services. A typical scenario for using this protocol is to manage a purchase order from origination to completion when human interaction is necessary. As the workflow progresses through the steps from origination to completion, an individual such as an office manager might have to approve purchases over a certain amount of money. The office manager can use the protocol client to act on any of the workflow steps assigned to him/her, reassign that step to someone else if necessary, and also manage the overall workflow.

Protocol servers are servers which implement the **Web services** specified by this protocol. The protocol server also provides the infrastructure for workflows which are manipulated by those Web services.

Protocol Clients are anything that issue commands to the service via the Web service methods specified in this protocol.

A common usage of this protocol is to provide the ability for a human to interact with an assigned step in a workflow. For example, this can be accomplished from the protocol client by the following steps:

1. The protocol client sends a task request containing an identifier to the protocol server.

2. The service responds with information corresponding to the task with the matching identifier.

3. The protocol client sends a response request containing the information which would update the task on the workflow server.

4. The service responds with an empty response message.

The following is a diagram of this exchange.
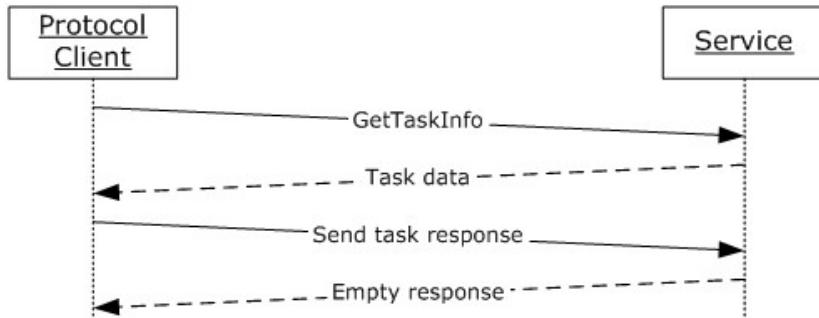
Figure 1: Message exchange used between protocol client and service to provide a user with an assigned step from a workflow

1.4 Relationship to Other Protocols

This protocol uses the **SOAP** message protocol for formatting request and response messages, as described in [SOAP1.1], [SOAP1.2/1] and [SOAP1.2/2]. It transmits those messages by using **HTTP**, as described in [RFC2616], or **HTTPS**, as described in [RFC2818].

The following diagram shows the underlying messaging and transport stack used by the protocol:
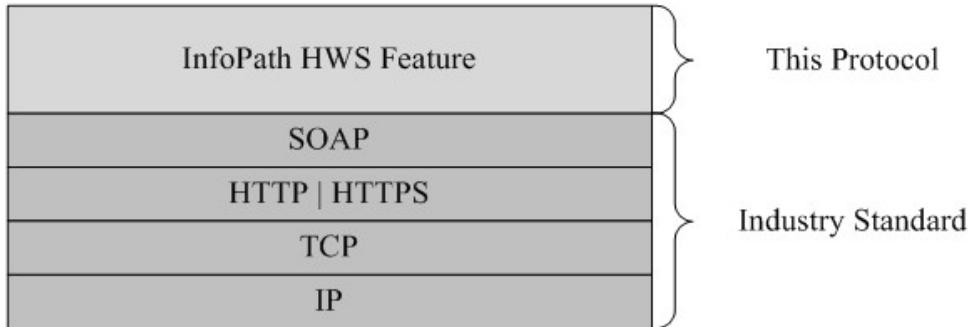
Figure 2: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a **site** that is identified by a **URL** that is known by protocol clients. The protocol server endpoint is formed by appending "hws/hwsservice/hwsservice.asmx" to the URL of the site, for example http://www.contoso.com/Repository/hws/hwsservice/hwsservice.asmx.

This protocol assumes that authentication has been performed by the underlying protocols.

1.6 Applicability Statement

This protocol is appropriate for use within a workflow environment when human interaction is necessary to complete specific steps within a workflow.

1.7 Versioning and Capability Negotiation

Supported Transports: This protocol uses multiple transports with SOAP (Simple Object Access Protocol), as specified in section [2.1].

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The protocol messages MUST be formatted as specified in [SOAP1.1] section 4. Protocol server faults MUST be returned using either Hypertext Transfer Protocol (HTTP) Status Codes, as specified in [RFC2616] section 10, or **SOAP faults**, as specified in [SOAP1.1] section 4.4.

The protocol server MUST support SOAP, as specified in [RFC2616], over Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS), as specified in [RFC2818] over **TCP/IP**.

2.2 Common Message Syntax

This section contains common definitions used by this protocol. The syntax of the definitions uses **XML Schema** as defined in [XMLSCHEMA1] and [XMLSCHEMA2], and Web Services Description Language as defined in [WSDL].

2.2.1 Namespaces

This specification specifies and references various **XML namespaces** using the mechanisms specified in [XMLNS] and [RFC2396]. Although this specification associates a specific XML Namespace prefix for each XML Namespace that is used, the choice of any particular XML Namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]
tns	http://microsoft.com/Biztalk2004/Hws/Hwsservice	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]

2.2.2 Messages

2.2.2.1 Fault

Protocol server faults MUST be returned using either HTTP Status Codes, as specified in [RFC2616] section 10, or SOAP faults, as specified in [SOAP1.1] section 4.4.

The details of a SOAP fault are as follows.

faultcode: The protocol server sends one of the following **SOAP fault codes** if errors are encountered.

Fault Code	Description
Client.Hws.InvalidParameter	Parameters passed by the client are not valid.

Fault Code	Description
Client.Hws.AccessDenied	Current user does not have permission to access data stored on the protocol server.
Client.Hws.Constraints	Current user does not satisfy the **Constraints** for performing an operation.
Server.Hws.Retrieve	Server fails to retrieve data requested by the protocol client.
Server.Hws.Activate	Server fails to activate an **action (1)**.
Server.Hws.Response	Server fails to send the response message.
Server.Hws.Engine	Server encountered a problem and the exact problem cannot be identified. This is a generic error code.

faultstring: A human readable text explaining the application-level fault.

2.2.3 Elements

This specification does not define any common XML Schema element definitions.

2.2.4 Complex Types

The following table summarizes the set of common XML Schema complex type definitions defined by this specification. XML Schema complex type definitions that are specific to a particular operation are specified with the operation.

Complex Type	Description
Activity	A data structure to represent an Activity.
ArrayOfString	An array of string objects.
ArrayOfActionInstance	An array of ActionInstance objects.
ActionInstance	A data structure to represent an Action Instance.
ArrayOfTask	An array of Task objects.
Task	A data structure to represent a Task.
ArrayOfTaskResponse	An array of TaskResponse objects.
TaskResponse	A data structure to represent a **task response**.
ArrayOfResourceProperty	An array of ResourceProperty objects.
ResourceProperty	A data structure to represent a **resource property**.
Reassignment	A data structure to store information about reassigning a Task to a different **actor**.

2.2.4.1 Activity

A data structure that stores information about an Activity.

```
<s:complexType name="Activity">
```

```
    <s:sequence>
      <s:element name="ActivityType" type="tns:ActivityType"/>
      <s:element name="ActionTypeID" type="tns:guid"/>
      <s:element name="ActivityModelTypeID" type="tns:guid"/>
      <s:element name="ActivationBlockID" type="tns:guid"/>
      <s:element name="ActionIndex" type="s:int"/>
      <s:element name="StepName" type="s:string" minOccurs="0"/>
      <s:element name="StepDescription" type="s:string" minOccurs="0"/>
      <s:element name="PreviousActionTypeID" type="tns:guid"/>
      <s:element name="IsDependentComposible" type="s:boolean"/>
      <s:element name="Name" type="s:string" />
      <s:element name="Description" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
```

ActivityType: Specifies the ActivityType of this Activity.

ActionTypeID: A **GUIDString** that specifies the action (1) for this Activity.

ActivityModelTypeID: A GUIDString that specifies the Activity Model for this Activity. The value of this element MUST be **NULL GUID** and MUST be ignored.

ActivationBlockID: A GUIDString that specifies the **activation block** in the Activity Model. The value of this element MUST be NULL GUID and MUST be ignored.

ActionIndex: An integer that specifies which **step** this Activity is in the Activity Model if this Activity is part of an Activity Model. The value of this element MUST be 0 and MUST be ignored.

StepName: The name of the Step if this Activity is part of an Activity Model. The value of this element MUST be NULL and MUST be ignored.

StepDescription: The description of the Step if this Activity is part of an Activity Model. The value of this element MUST be NULL and MUST be ignored.

PreviousActionTypeID: A GUIDString that specifies the action (1) of the parent Action Instance. The value of this element MUST be NULL GUID and MUST be ignored.

IsDependentComposible: Specifies whether this Activity can be used in a **dependent composition**. The value of this element MUST be false and MUST be ignored.

Name: The name of the Activity.

Description: The description of the Activity.

2.2.4.2   ArrayOfString

An array of string elements.

```
  <s:complexType name="ArrayOfString">
    <s:sequence>
      <s:element name="string" type="s:string" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    </s:sequence>
  </s:complexType>
```

string: An array element of type s:string.

2.2.4.3 ArrayOfActionInstance

An array of [ActionInstance] elements.

```
<s:complexType name="ArrayOfActionInstance">
  <s:sequence>
    <s:element name="ActionInstance" type="tns:ActionInstance" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

ActionInstance: An array element of type ActionInstance.

2.2.4.4 ActionInstance

A data structure that stores information about an ActionInstance.

```
<s:complexType name="ActionInstance">
  <s:sequence>
    <s:element name="ActivityFlowID" type="tns:guid"/>
    <s:element name="ActionTypeID" type="tns:guid"/>
    <s:element name="ActionInstanceID" type="tns:guid"/>
    <s:element name="PreviousActionInstanceID" type="tns:guid"/>
    <s:element name="PreviousTaskID" type="tns:guid"/>
    <s:element name="ActivityModelTypeID" type="tns:guid"/>
    <s:element name="ActivityModelIndex" type="s:int"/>
    <s:element name="ActivityModelInstanceID" type="tns:guid"/>
    <s:element name="Name" type="s:string" />
    <s:element name="ActivityModelName" type="s:string" minOccurs="0"/>
    <s:element name="ActionInstanceTitle" type="s:string" />
    <s:element name="Initiator" type="s:string" />
    <s:element name="IsDependent" type="s:boolean"/>
    <s:element name="Status" type="tns:ActionStatus"/>
    <s:element name="StartTime" type="s:dateTime"/>
    <s:element name="EndTime" type="s:dateTime"/>
    <s:element name="Tasks" type="tns:ArrayOfTask" minOccurs="0"/>
    <s:element name="ActionProperties" type="tns:ArrayOfResourceProperty" minOccurs="0"/>
    <s:element name="ChildActionInstances" type="tns:ArrayOfActionInstance" minOccurs="0"/>
  </s:sequence>
</s:complexType>
```

ActivityFlowID: A GUIDString that identifies the **activity flow** that this ActionInstance belongs to.

ActionTypeID: A GUIDString that identifies the action (1) of this ActionInstance.

ActionInstanceID: A GUIDString that identifies this ActionInstance.

PreviousActionInstanceID: A GUIDString that identifies the parent ActionInstance. If this is a NULL GUID, this ActionInstance is the root ActionInstance of the activity flow and has no parent. A parent ActionInstance is an ActionInstance in an activity flow that precedes the current ActionInstance and is specified when adding an action (1) to an activity flow using the method AddActionToActivityFlow.

PreviousTaskID: A GUIDString that identifies the Task sent by the parent ActionInstance which has led to the creation of this ActionInstance.

ActivityModelTypeID: A GUIDString that identifies the ActivityModel that the action (1) of this ActionInstance belongs to. The value of this element MUST be ignored.

ActivityModelIndex: The step identifier of the action (1) of this ActionInstance within the ActivityModel. The value of this element MUST be ignored.

ActivityModelInstanceID: A GUIDString that identifies the ActivityModel. The value of this element MUST be ignored.

Name: The name of the action (1) identified by ActionTypeID.

ActivityModelName: The name of the ActivityModel identified by ActivityModelTypeID. The value of this element MUST be ignored.

ActionInstanceTitle: A string that can be set by the user to give each ActionInstance a human-readable title.

Initiator: The name of the Actor that created this ActionInstance.

IsDependent: A Boolean that indicates whether this instance has been added to the ActivityFlow through Dependent Composition. The value of this element MUST be ignored.

Status: The current status of the ActionInstance.

StartTime: The time stamp expressed as **Coordinated Universal Time (UTC)** indicating when the ActionInstance is created.

EndTime: The time stamp expressed as **UTC** indicating the last time a message was received for this ActionInstance.

Tasks: An array of Tasks created by this ActionInstance.

ActionProperties: An array of ResourceProperties that are used to store any information needed to implement the logic inside the action (1). Each property has a name, type, description and value. It directly corresponds to one of the ActionProperties/Property nodes in the **activation message**.

ChildActionInstances: Array of child ActionInstances.

2.2.4.5 ArrayOfTask

An array of Task objects.

```
<s:complexType name="ArrayOfTask">
  <s:sequence>
    <s:element name="Task" type="tns:Task" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

Task: An array element of type Task.

2.2.4.6 Task

A data structure that stores information about a Task.

```
<s:complexType name="Task">
```

```
<s:sequence>
  <s:element name="Initiator" type="s:string" />
  <s:element name="Target" type="s:string" />
  <s:element name="TargetXPath" type="s:string" minOccurs="0"/>
  <s:element name="TaskID" type="tns:guid"/>
  <s:element name="TaskDescription" type="s:string" minOccurs="0"/>
  <s:element name="ActivityFlowID" type="tns:guid"/>
  <s:element name="ActionInstanceID" type="tns:guid"/>
  <s:element name="TaskTimeStamp" type="s:dateTime"/>
  <s:element name="CurrentStatus" type="tns:TaskStatus"/>
  <s:element name="CurrentPercentageComplete" type="s:int"/>
  <s:element name="Responses" type="tns:ArrayOfTaskResponse" minOccurs="0"/>
  <s:element name="TargetNamespaceUri" type="s:string"/>
  <s:element name="TaskProperties" type="tns:ArrayOfResourceProperty" minOccurs="0"/>
  <s:element name="Reassignment" type="tns:Reassignment"/>
</s:sequence>
</s:complexType>
```

Initiator: The name of the Actor that created this Task.

Target: The name of the Actor the Task is assigned to.

TargetXPath: An **XPath expression** used to locate the **XML** element that contains the Target in the **task message**. The value of this element MUST be ignored.

TaskID: A GUIDString that identifies this Task.

TaskDescription: A string that specifies the purpose of this Task.

ActivityFlowID: A GUIDString that identifies the Activity Flow this Task belongs to.

ActionInstanceID: A GUIDString that identifies the Action Instance that created this Task.

TaskTimeStamp: The time stamp expressed as UTC when this Task is created.

CurrentStatus: The current status of this Task.

CurrentPercentageComplete: Specifies the progress of the Task by assigning a percentage number for work that has been done.

Responses: An array of TaskResponses.

TargetNamespaceUri: The namespace URI of the Task Message.

TaskProperties: An array of ResourceProperties that stores application data for this Task.

Reassignment: A Reassignment object that links the old and the new Tasks.

2.2.4.7   ArrayOfTaskResponse

An array of TaskResponses.

```
<s:complexType name="ArrayOfTaskResponse">
  <s:sequence>
    <s:element name="TaskResponse" type="tns:TaskResponse" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
```

```
        </s:complexType>
```

TaskResponse: An array element of type TaskResponses.

2.2.4.8   TaskResponse

A data structure to stores properties of a Task Response.

```
<s:complexType name="TaskResponse">
  <s:sequence>
    <s:element name="ResponseTimeStamp" type="s:dateTime"/>
    <s:element name="Status" type="tns:TaskStatus"/>
    <s:element name="PercentageComplete" type="s:int"/>
    <s:element name="Responder" type="s:string" />
    <s:element name="ResponseProperties" type="tns:ArrayOfResourceProperty" minOccurs="0"/>
  </s:sequence>
</s:complexType>
```

ResponseTimeStamp: The time stamp expressed as UTC when the Task Response is received.

Status: The current status of the Task.

PercentageComplete: A percentage number that indicates the progress towards the task completion.

Responder: The name of the Actor that sent the response.

ResponseProperties: An array of ResourceProperties.

2.2.4.9   ArrayOfResourceProperty

An array of ResourceProperty objects.

```
<s:complexType name="ArrayOfResourceProperty">
  <s:sequence>
    <s:element name="ResourceProperty" type="tns:ResourceProperty" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

ResourceProperty: An array element of type ResourceProperty.

2.2.4.10   ResourceProperty

A data structure that holds application data that is passed between protocol client and protocol server. Data stored in Resource Properties are not interpreted by the protocol server. The application implementer decides how to interpret and manipulate this data.

```
<s:complexType name="ResourceProperty">
  <s:sequence>
    <s:element name="Name" type="s:string" />
    <s:element name="Type" type="s:string" />
    <s:element name="Description" type="s:string" minOccurs="0"/>
    <s:element name="Value" type="s:string" />
```

```
      </s:sequence>
    </s:complexType>
```

Name: Name of the property.

Type: Type of the property.

Description: Description of the property.

Value: Value of the property.

2.2.4.11 Reassignment

A data structure that links the current Task to the reassigned Task. When a Task is reassigned, delegated, or escalated to another Actor, a new Task is created to reassign, delegate, or escalate. The Reassignment data structure maintains the relationships between the new and the old Tasks. It allows for navigation up and down a Task reassignment chain.

```
  <s:complexType name="Reassignment">
    <s:sequence>
      <s:element name="PreviousTaskID" type="tns:guid"/>
      <s:element name="NextTaskID" type="tns:guid"/>
      <s:element name="FirstTaskID" type="tns:guid"/>
      <s:element name="LastTaskID" type="tns:guid"/>
    </s:sequence>
  </s:complexType>
```

PreviousTaskID: A GUIDString that identifies the previous Task.

NextTaskID: A GUIDString that identifies the next Task.

FirstTaskID: A GUIDString that identifies the first Task in the reassignment chain.

LastTaskID: A GUIDString that identifies the last Task in the reassignment chain.

2.2.5 Simple Types

The following table summarizes the set of common XML Schema simple type definitions defined by this specification. XML Schema simple type definitions that are specific to a particular operation are specified with the operation.

Values for all string types defined in Messages and Protocol Details MUST be **Unicode** strings, as specified in [RFC2781].

Simple Type	Description
ActivityType	Indicates the type of an **activity**.
ActionStatus	Indicates the current status of an **action instance**.
TaskStatus	Indicates the current status of a **task**.
Guid	A GUIDString.

2.2.5.1 ActivityType

Specifies the type of an Activity.

```
<s:simpleType name="ActivityType">
  <s:restriction base="s:string">
    <s:enumeration value="Action"/>
    <s:enumeration value="ActivityModel"/>
  </s:restriction>
</s:simpleType>
```

The following table specifies the allowable values for ActivityType:

Value	Meaning
Action	The Activity is an action (1).
ActivityModel	The Activity is an **activity model**.

2.2.5.2 ActionStatus

Specifies the status of an ActionInstance.

```
<s:simpleType name="ActionStatus">
  <s:restriction base="s:string">
    <s:enumeration value="Start"/>
    <s:enumeration value="Finish"/>
    <s:enumeration value="Abort"/>
  </s:restriction>
</s:simpleType>
```

The following table specifies the allowable values for ActionStatus:

Value	Meaning
Start	The ActivityInstance has started. The ActivityInstance enters in the Start state when it is created and stays in the Start state until it finishes execution.
Finish	The ActivityInstance has finished. The ActivityInstance enters the Finish state when the ActivityInstance receives a **finish message**.
Abort	The ActivityInstance was aborted. The ActivityInstance enters the Abort state when it receives an **interrupt message**.

2.2.5.3 TaskStatus

Specifies the status of a Task. Status values for a Task are generally assigned by the protocol client with the exception of "Interrupted" and "Cancelled", which are assigned by the protocol server.

```
<s:simpleType name="TaskStatus">
  <s:restriction base="s:string">
    <s:enumeration value="NotStarted"/>
    <s:enumeration value="InProgress"/>
    <s:enumeration value="Completed"/>
```

```
      <s:enumeration value="Waiting"/>
      <s:enumeration value="Deferred"/>
      <s:enumeration value="Accepted"/>
      <s:enumeration value="Tentative"/>
      <s:enumeration value="Declined"/>
      <s:enumeration value="Interrupted"/>
      <s:enumeration value="Cancelled"/>
    </s:restriction>
  </s:simpleType>
```

The following table specifies the allowable values for TaskStatus.

Value	Meaning
NotStarted	The Task is created.
InProgress	The Task is currently in progress.
Completed	The Task has finished.
Waiting	The Task is waiting for other work to be done before continuing.
Deferred	The Task is deferred.
Accepted	The Task was accepted by its **target**.
Tentative	The Task was tentatively accepted by its Target.
Declined	The Task was declined by its Target.
Interrupted	The Task is interrupted if the ActionInstance is aborted before the Task is Completed, Deferred or Declined.
Cancelled	The Task is cancelled if the ActionInstance is finished before the Task is Completed, Deferred or Declined.

2.2.5.4   guid

Specifies the string representation of a **globally unique identifier (GUID)** in the form of a GUIDString.

```
<s:simpleType name="guid">
  <s:restriction base="s:string">
    <s:pattern value="[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-
F]{12}"/>
  </s:restriction>
</s:simpleType>
```

2.2.6   Attributes

This specification does not define any common XML Schema attribute definitions.

2.2.7   Groups

This specification does not define any common XML Schema group definitions.

2.2.8   Attribute Groups

This specification does not define any common XML Schema attribute group definitions.

3   Protocol Details

3.1   HwsServiceSoap Server Details

The protocol server enables the protocol client to connect to a **workflow system** that allows human interaction with an automated business process. Activity Flows are dynamically constructed as users create new Action Instances and respond to Tasks.

The protocol server is stateless. All methods on the Web service can be called in random order. However, there is application state maintained in the Workflow System. Thus a certain call order is implied. The following state diagram shows a typical call pattern that a protocol client would use.

Figure 3: A typical call pattern used by a protocol client

1. The protocol client starts by calling GetActivityList to obtain an array of Actions that the current user can **Activate**.

2. The protocol client chooses one action (1) and calls AddActionToActivityFlow to begin a new Activity Flow.

3. Once the Activity Flow is created, the protocol client can call methods to retrieve information about the Activity Flow and objects in the Activity Flow. These methods are GetActivityFlowInfo, GetActionInstance, GetTaskInfo.

4. If a Task is created by an Action Instance in the Activity Flow, the protocol client can respond to the Task by calling GetTaskMessage followed by SendTaskResponse.

5. The protocol client can extend the Activity Flow by adding new Action Instances. To add a new Action Instance to the Activity Flow, the protocol client calls GetActivityList followed by AddActionToActivityFlow.

6. If all Action Instances in the Activity Flow have finished, the Activity Flow becomes inactive. GetActivityFlowInfo, GetActionInstance, GetTaskInfo and GetTaskMessage can be called on inactive Activity Flows, but SendTaskResponse cannot.

7. The protocol client can reactivate the Activity Flow by adding new Action Instances. To add a new Action Instance to the Activity Flow, the protocol client calls GetActivityList followed by AddActionToActivityFlow.

3.1.1 Abstract Data Model

This section specifies a conceptual model of possible data organization that a server implementation maintains to participate in this protocol. The specified organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that specified in this document.

The MS-IPHWS protocol server keeps the following two types of data in a permanent data storage: Workflow definitions in the form of Actions and Workflow instances in the form of Activity Flows.

A **tracking system** stores information about Activity Flows as they are created and extended. The Tracking System can organize tracking data into separate database tables to track information about Activity Flows, Action Instances, Tasks and Actors. It can also create additional tables to track the relationship between Activity Flows, Action Instances, Tasks and Actors. The Tracking System can also perform queries to return tracked information about each Activity Flow, Action Instance, Task and Actors.

3.1.2 Timers

None.

3.1.3 Initialization

When starting the Web service, the protocol server initializes the Workflow System to ensure that the system is ready to accept requests from protocol clients.

3.1.4 Message Processing Events and Sequencing Rules

The following table lists the web methods available on the protocol server in alphabetical order.

Operation	Description
AddActionToActivityFlow	Creates an Action Instance to either begin a new Activity Flow or extend an existing Activity Flow.
GetActionInstance	Retrieves information about an Action Instance from the Tracking System where the current user is a **Participant**.
GetActivityFlowInfo	Retrieves information about an Activity Flow from the Tracking System where the current user is a Participant.
GetActivityList	Returns the list of Actions that the current user can Activate.
GetTaskInfo	Retrieves information about a Task from the Tracking System where the current user is a Participant.
GetTaskMessage	Returns the Task Message for a Task.
SendTaskResponse	Sends a Task Response to a Task.

3.1.4.1 AddActionToActivityFlow

This operation is used to create a new Action Instance to either begin a new Activity Flow or extend an existing Activity Flow.

```
<wsdl:operation name="AddActionToActivityFlow">
    <wsdl:input message="AddActionToActivityFlowSoapIn" />
    <wsdl:output message="AddActionToActivityFlowSoapOut" />
</wsdl:operation>
```
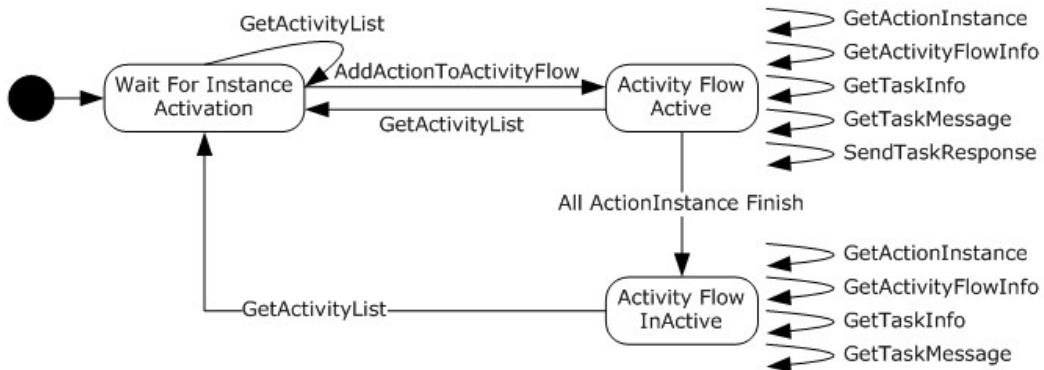
The protocol client sends an AddActionToActivityFlowSoapIn request message and the protocol server responds with an AddActionToActivityFlowSoapOut response message, as follows:

1. If activityFlowID is NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. If an Activity Flow identified by activityFlowID exists, the protocol server MUST check if the current user is a Participant in the Activity Flow. If the user is not a participant, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

3. If actionParameters is NULL or is an empty element, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

4. If parentActionInstanceID is NULL GUID, then this is a request to create a new Activity Flow, in which case an Activity Flow MUST NOT exist with the same identifier as activityFlowID. If such activity flow already exists, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

5. If parentActionInstanceID is NULL GUID and parentTaskID is not NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

6. If parentActionInstanceID is not NULL GUID, this is a request to extend an existing Activity Flow and an Activity Flow MUST exist with the same identifier as activityFlowID. If this Activity Flow does not exist, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

7. The parameters document passed in through actionParameters.ParametersDoc MUST be a valid XML document that satisfies the Activation Message schema defined in section 7.1. If the parameters document is not a valid XML document, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.Constraints". If the parameters document does not satisfy the Activation Message schema, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Activate ".

8. The protocol server finds the ActionTypeID element in the parameters document and retrieves the action (1) identified by the value of this element.

9. The current user is evaluated against Constraints created by the application implementer. If the current user does not pass the constraints check, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.Constraints".

10. The protocol server MUST create a new ActionInstance by activating the action (1) identified by the ActionTypeID in the parameters document.

11. If an Activity Flow with an identifier the same as an activityFlowID does not exist, the protocol server MUST create an Activity Flow with activityFlowID and add the ActionInstance as the first ActionInstance in the Activity Flow.

12. If an Activity Flow with an identifier the same as an activityFlowID already exists, the protocol server MUST add the ActionInstance to the existing flow.

13. The protocol server MUST send back an AddActionToActivityFlowSoapOut response message containing the GUIDString that identifies the newly created ActionInstance.

3.1.4.1.1 Messages

Specifies the **SOAP messages** sent between the protocol client and the protocol server when the protocol client calls the AddActionToActivityFlow method on the Web service.

3.1.4.1.1.1 AddActionToActivityFlowSoapIn

This message is sent from the protocol client to the protocol server to perform the AddActionToActivityFlow operation on the protocol server.

The **SOAP action** value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/AddActionToActivityFlow
```

The **SOAP body** contains an AddActionToActivityFlow element.

3.1.4.1.1.2 AddActionToActivityFlowSoapOut

This message is sent from the protocol server to the protocol client in response to the request message AddActionToActivityFlowSoapIn.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/AddActionToActivityFlow
```

The SOAP body contains an AddActionToActivityFlowResponse element.

3.1.4.1.2 Elements

3.1.4.1.2.1 AddActionToActivityFlow

Specifies the name and the parameters for the Web service method AddActionToActivityFlow.

```
<s:element name="AddActionToActivityFlow">
  <s:complexType>
    <s:sequence>
      <s:element name="activityFlowID" type="tns:guid"/>
      <s:element name="parentActionInstanceID" type="tns:guid"/>
      <s:element name="parentTaskID" type="tns:guid"/>
      <s:element name="isDependentComposition" type="s:boolean"/>
      <s:element name="actionParameters" type="tns:ActionParameters" />
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

activityFlowID: A GUIDString that identifies the Activity Flow to which the new ActionInstance is to be added.

parentActionInstanceID: A GUIDString that identifies the parent ActionInstance.

parentTaskID: A GUIDString that identifies the Task sent by the parent ActionInstance that led to the creation of this ActionInstance.

isDependentComposition: A Boolean that specifies whether or not this is a dependent composition. The value of this element MUST be false and MUST be ignored.

actionParameters: An instance of the ActionParameters data structure.

actingUser: When using **trusted authentication**, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.1.2.2 AddActionToActivityFlowResponse

Specifies the response the protocol server sends after processing the AddActionToActivityFlow Web service call. When the Web service call successfully returns, a GUIDString that identifies the newly created ActionInstance is returned to the protocol client.

```
<s:element name="AddActionToActivityFlowResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="ActionInstanceID" type="tns:guid"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

ActionInstanceID: A GUIDString that identifies the newly created ActionInstance. The value of this element MUST NOT be NULL GUID.

3.1.4.1.3 Complex Types

3.1.4.1.3.1 ActionParameters

A data structure that stores the parameters needed to create a new ActionInstance.

```
<s:complexType name="ActionParameters">
  <s:sequence>
    <s:element name="Action" type="tns:Activity" />
    <s:element name="XsdDoc" type="s:string" minOccurs="0"/>
    <s:element name="TaskSchemas" type="tns:ArrayOfString" minOccurs="0"/>
    <s:element name="Targets" type="tns:ArrayOfTarget" minOccurs="0"/>
    <s:element name="ParametersDoc" type="s:string" />
    <s:element name="ActionInstanceTitle" type="s:string" />
    <s:element name="IsOverridable" type="s:boolean"/>
  </s:sequence>
</s:complexType>
```

Action: The Activity object for which the rest of the parameters apply.

XsdDoc: An XML Schema document that specifies the schema of the Activation Message. The value of this element MUST be NULL and MUST be ignored.

TaskSchemas: An array of XML Schema documents; each specifies the schema of one Task Message that the action (1) specifies. The value of this element MUST be NULL and MUST be ignored.

Targets: An array of XPath expressions which the protocol client can use to identify the Target element in the Activation Message. The value of this element MUST be NULL and MUST be ignored.

ParametersDoc: An XML instance document whose schema is defined by the Activation Message schema in section 7.1.

ActionInstanceTitle: A user friendly string to differentiate each ActionInstance of the same action (1).

IsOverridable: This parameter is reserved for future use. The value of this element MUST be false and MUST be ignored.

3.1.4.1.3.2 ArrayOfTarget

An array of Target objects.

```
<s:complexType name="ArrayOfTarget">
  <s:sequence>
    <s:element name="Target" type="tns:Target" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

Target: An array element of type Target.

3.1.4.1.3.3 Target

Represents an XPath expression used to identify an element in the Activation Message that contains the Target of a Task.

```
<s:complexType name="Target">
  <s:sequence>
    <s:element name="ParameterName" type="s:string" />
    <s:element name="InstanceXPath" type="s:string" />
    <s:element name="Min" type="s:int"/>
    <s:element name="Max" type="s:int"/>
  </s:sequence>
</s:complexType>
```

ParameterName: Specifies the name of the node identified by the InstanceXPath property in an Activation Message.

InstanceXPath: Specifies the XPath to node that can be marked as target node within an activation message.

Min: Specifies the minimum number of occurrences of the node identified by the InstanceXPath property.

Max: Specifies the maximum number of occurrences of the node identified by the InstanceXPath property.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 GetActionInstance

This operation is used to retrieve information about an Action Instance from the Tracking System.

```
<wsdl:operation name="GetActionInstance">
    <wsdl:input message="GetActionInstanceSoapIn" />
    <wsdl:output message="GetActionInstanceSoapOut" />
</wsdl:operation>
```

The protocol client sends a GetActionInstanceSoapIn request message and the protocol server responds with a GetActionInstanceSoapOut response message, as follows:

1. If actionInstanceID is NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. The protocol server retrieves the ActionInstance identified by actionInstanceID from the Tracking System. If the protocol server fails to retrieve the information for the specified Action Instance, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

3. The protocol server MUST verify that the current user is a Participant in the Activity Flow that this Action Instance belongs to. If the current user is not a Participant, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

4. The protocol server MUST send the GetActionInstanceSoapOut message containing the ActionInstance back to the protocol client.

3.1.4.2.1 Messages

3.1.4.2.1.1 GetActionInstanceSoapIn

SOAP message sent by the protocol client to the protocol server to call the GetActionInstance Web service method.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActionInstance
```

The SOAP body contains a GetActionInstance element.

3.1.4.2.1.2 GetActionInstanceSoapOut

SOAP message sent by the protocol server to the protocol client in response to the GetActionInstance Web service call.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActionInstance
```

The SOAP body contains a GetActionInstanceResponse element.

3.1.4.2.2 Elements

3.1.4.2.2.1 GetActionInstance

Specifies the name and parameters of the Web service method GetActionInstance.

```
<s:element name="GetActionInstance">
  <s:complexType>
    <s:sequence>
      <s:element name="actionInstanceID" type="tns:guid"/>
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

actionInstanceID: A GUIDString that identifies the ActionInstance the protocol client is requesting.

actingUser: When using trusted authentication, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.2.2.2 GetActionInstanceResponse

Specifies the response the protocol server sends after processing the GetActionInstance Web service call. Upon the completion of the Web service call, the protocol server returns an ActionInstance object that contains properties of the requested ActionInstance.

```
<s:element name="GetActionInstanceResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="ActionInstance" type="tns:ActionInstance" />
    </s:sequence>
  </s:complexType>
</s:element>
```

ActionInstance: An object of type ActionInstance. The value of this element MUST NOT be NULL.

3.1.4.2.3 Complex Types

None.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 GetActivityFlowInfo

Retrieves information about an Activity Flow from the Tracking System.

```
<wsdl:operation name="GetActivityFlowInfo">
    <wsdl:input message="GetActivityFlowInfoSoapIn" />
    <wsdl:output message="GetActivityFlowInfoSoapOut" />
</wsdl:operation>
```

The protocol client sends a GetActivityFlowInfoSoapIn request message and the protocol server responds with a GetActivityFlowInfoSoapOut response message, as follows:

1. If activityFlowID is NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. The protocol server retrieves the ActivityFlow identified by activityFlowID from the Tracking System. If the protocol server fails to retrieve the information, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

3. The protocol server MUST verify that the current user is a Participant in the Activity Flow identified by activityFlowID. If the current user is not a Participant, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

4. The protocol server MUST send the GetActivityFlowInfoSoapOut message containing the ActivityFlow back to the protocol client.

3.1.4.3.1 Messages

3.1.4.3.1.1 GetActivityFlowInfoSoapIn

The request message sent by the protocol client to the Web service to call the GetActivityFlowInfo web method.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityFlowInfo
```

The SOAP body contains a GetActivityFlowInfo element.

3.1.4.3.1.2 GetActivityFlowInfoSoapOut

The response message sent by the protocol server in response to the invocation of the GetActivityFlowInfo web method.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityFlowInfo
```

The SOAP body contains a GetActivityFlowInfoResponse element.

3.1.4.3.2 Elements

3.1.4.3.2.1 GetActivityFlowInfo

This element is used in the request message to call the GetActivityFlowInfo web method on the protocol server.

```
<s:element name="GetActivityFlowInfo">
  <s:complexType>
    <s:sequence>
      <s:element name="activityFlowID" type="tns:guid"/>
      <s:element name="detailLevel" type="tns:ActivityFlowDetailLevel"/>
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

activityFlowID: The GUIDString that identifies the ActivityFlow whose information the protocol client is requesting.

detailLevel: The level of details the protocol client needs to know about the ActivityFlow. The value of this element MUST be ActionInstanceLevel.

actingUser: When using trusted authentication, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.3.2.2 GetActivityFlowInfoResponse

This element is used in the response message the protocol server sends to the protocol client in response to the GetActivityFlowInfo Web service call.

```
<s:element name="GetActivityFlowInfoResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="ActivityFlowTrackingInfo" type="tns:ActivityFlow" />
    </s:sequence>
  </s:complexType>
</s:element>
```

ActivityFlowTrackingInfo: An ActivityFlow object that contains information about the Activity Flow. The value of this element MUST NOT be NULL.

3.1.4.3.3 Complex Types

3.1.4.3.3.1 ActivityFlow

A data structure that stores properties of an Activity Flow.

```
<s:complexType name="ActivityFlow">
  <s:sequence>
    <s:element name="DetailLevel" type="tns:ActivityFlowDetailLevel"/>
    <s:element name="StatInfo" type="tns:ActivityFlowStat" minOccurs="0"/>
    <s:element name="ActivityFlowID" type="tns:guid"/>
    <s:element name="ActivityFlowDescription" type="s:string" minOccurs="0"/>
    <s:element name="Status" type="tns:ActivityFlowStatus"/>
    <s:element name="RootActionInstances" type="tns:ArrayOfActionInstance" />
    <s:element name="Initiator" type="s:string" />
    <s:element name="AllIgnoring" type="s:boolean"/>
    <s:element name="StartTimeStamp" type="s:dateTime"/>
    <s:element name="LastModifiedTimeStamp" type="s:dateTime"/>
    <s:element name="ActivityFlowProperties" type="tns:ArrayOfResourceProperty"
minOccurs="0"/>
  </s:sequence>
</s:complexType>
```

DetailLevel: Specifies the verbose level of the information included.

StatInfo: Stores the current statistics about the Activity Flow.

ActivityFlowID: A GUIDString that identifies the Activity Flow whose information is being returned.

ActivityFlowDescription: The description of the Activity Flow.

Status: The current status of the Activity Flow.

RootActionInstances: An array that contains the first ActionInstance in this Activity Flow. There MUST be only one root ActionInstance for each Activity Flow.

Initiator: The **Initiator** of the root ActionInstance.

AllIgnoring: A Boolean value. True indicates that all Actors involved in the Activity Flow have chosen to ignore it; otherwise, false. The value of this element MUST be ignored.

StartTimeStamp: The time stamp expressed as UTC, indicating when the root ActionInstance was created.

LastModifiedTimeStamp: The time stamp expressed as UTC, indicating when the last update to the Activity Flow was received.

ActivityFlowProperties: An array of all ResourceProperties defined for this Activity Flow.

3.1.4.3.3.2 ActivityFlowStat

A data structure to store statistic information about an ActivityFlow.

```
<s:complexType name="ActivityFlowStat">
  <s:sequence>
    <s:element name="DetailLevel" type="tns:ActivityFlowDetailLevel"/>
    <s:element name="ActionInstanceCount" type="s:int"/>
```

```
        <s:element name="AdHocActionInstanceCount" type="s:int"/>
        <s:element name="ActivityModelCount" type="s:int"/>
        <s:element name="LastActionName" type="s:string" minOccurs="0"/>
        <s:element name="LastActionInitiator" type="s:string" minOccurs="0"/>
        <s:element name="LastActionTargets" type="tns:ArrayOfString" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
```

DetailLevel: Specifies how much information to return to the protocol client in a retrieval function call.

ActionInstanceCount: The total number of ActionInstances in the ActivityFlow.

AdHocActionInstanceCount: The number of ActionInstances added to the ActivityFlow via **ad hoc composition**.

ActivityModelCount: The number of Activity Model instances in the ActivityFlow.

LastActionName: The name of the ActionInstances that was last added to the ActivityFlow.

LastActionInitiator: The Initiator of the last ActionInstances that was added to the ActivityFlow.

LastActionTargets: The list of Targets for the last ActionInstances that was added to the ActivityFlow.

3.1.4.3.4 Simple Types

3.1.4.3.4.1 ActivityFlowDetailLevel

Specifies the level of detail that the Tracking System can return about an Activity Flow.

```
  <s:simpleType name="ActivityFlowDetailLevel">
    <s:restriction base="s:string">
      <s:enumeration value="ActionInstanceLevel"/>
    </s:restriction>
  </s:simpleType>
```

The following table specifies the allowable values for ActivityFlowDetailLevel:

Value	Meaning
ActionInstanceLevel	Indicates that the ActivityFlow returned by the Tracking System includes information about the Activity Flow and any ActionInstances in the Activity Flow. It does not include information about the Tasks Associated with the ActionInstances in the Activity Flow.

3.1.4.3.4.2 ActivityFlowStatus

Indicates the status of an Activity Flow.

```
  <s:simpleType name="ActivityFlowStatus">
    <s:restriction base="s:string">
      <s:enumeration value="Active"/>
      <s:enumeration value="InActive"/>
```

```
      <s:enumeration value="Complete"/>
    </s:restriction>
  </s:simpleType>
```

The following table specifies the allowable values for ActivityFlowStatus:

Value	Meaning
Active	The Activity Flow is Active when at least one Action Instance in the Activity Flow is not Finished or Aborted.
InActive	The Activity Flow is inactive. An Activity Flow is inactive if all Action Instances are Finished or Aborted.
Complete	The Activity Flow is finished. A completed Activity Flow cannot be extended.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.4.4 GetActivityList

This operation is used to return a list of Activities that the current user can Activate.

```
<wsdl:operation name="GetActivityList">
    <wsdl:input message="GetActivityListSoapIn" />
    <wsdl:output message="GetActivityListSoapOut" />
</wsdl:operation>
```

The protocol client sends a GetActivityListSoapIn request message and the protocol server responds with a GetActivityListSoapOut response message, as follows:

1. If activityFlowID is NULL GUID, the protocol server MUST return the list of all Activities available for **activation** to the protocol client.

2. If parentActionInstanceID is NULL GUID and parentTaskID is not NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

3. If parentActionInstanceID is NULL GUID and activityFlowID is a **GUID** which is not equal to an existing instance of an activityFlowID, the protocol server MUST return the list of all Activities available for activation to the protocol client.

4. If parentActionInstanceID is a NULL GUID and activityFlowID is not NULL GUID and is equal to an existing instance of an activityFlowID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

5. If activityFlowID is not NULL GUID and is not equal to an existing instance of an activityFlowID and the parentActionInstanceID is not NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

6. If activityFlowID is not NULL GUID and is equal to an existing instance of an activityFlowID, and parentTaskID is not NULL GUID and is not equal to an existing instance of a Task, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

7. If activityFlowID is not NULL GUID and is equal to an existing instance of an activityFlowID, and parentActionInstanceID is not NULL GUID and is not equal to an existing instance of an activityFlowID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

8. The protocol server MUST verify that the current user is a Participant in the Activity Flow identified by activityFlowID. If the current user is not a Participant, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

9. If activityFlowID is not NULL GUID and is equal to an existing instance of an activityFlowID and parentActionInstanceID is not NULL GUID and is equal to an existing instance of an activityFlowID, the protocol server MUST return the list of Activities identified by the activityFlowID.

10. The protocol server MUST return a list of Activities available for Activation by the current user, or NULL if there are no available Activities to the protocol client.

3.1.4.4.1 Messages

3.1.4.4.1.1 GetActivityListSoapIn

The request message sent by the protocol client to the protocol server to call the GetActivityList Web service call.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityList
```

The SOAP body contains a GetActivityList element.

3.1.4.4.1.2 GetActivityListSoapOut

GetActivityListSoapOut is the response message sent by the protocol server to the protocol client upon successful completion of the GetActivityList Web service call.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityList
```

The SOAP body contains a GetActivityListResponse element.

3.1.4.4.2 Elements

3.1.4.4.2.1 GetActivityList

This element is used in the GetActivityListSoapIn message to specify parameters for the
GetActivityList Web service call.

```
<s:element name="GetActivityList">
  <s:complexType>
    <s:sequence>
      <s:element name="activityFlowID" type="tns:guid"/>
      <s:element name="parentActionInstanceID" type="tns:guid"/>
      <s:element name="parentTaskID" type="tns:guid"/>
      <s:element name="target" type="s:string" minOccurs="0"/>
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

activityFlowID: A GUIDString that identifies the Activity Flow to which the ActionInstance
identified by parentActionInstanceID belongs.

parentActionInstanceID: A GUIDString that identifies the previous Action Instance.

parentTaskID: A GUIDString that identifies the Task sent by the Action Instance identified by
parentActionInstanceID.

target: The name of the Actor who is going to be the Target of the next ActionInstance. The value
of this element MUST be ignored.

actingUser: When using Trusted Authentication, the name of the user who is participating in the
Activity Flow. The value of this element MUST be ignored.

3.1.4.4.2.2 GetActivityListResponse

This element is used in the GetActivityListSoapOut message to specify the return value of the
GetActivityList web method call.

```
<s:element name="GetActivityListResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="ExecutableActions" type="tns:Activity" minOccurs="0"
maxOccurs="unbounded"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

ExecutableActions: An array of Activity objects that can be activated by the current user.

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 GetTaskInfo

Retrieves information about a Task from the Tracking System.

```
<wsdl:operation name="GetTaskInfo">
    <wsdl:input message="GetTaskInfoSoapIn" />
    <wsdl:output message="GetTaskInfoSoapOut" />
</wsdl:operation>
```

The protocol client sends a GetTaskInfoSoapIn request message and the protocol server responds with a GetTaskInfoSoapOut response message, as follows:

1. If taskID is NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. The protocol server retrieves the Task identified by taskID from the Tracking System. If the protocol server fails to retrieve the Task, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

3. The protocol server MUST verify that the current user is a Participant in the ActivityFlow identified by the ActivityFlowID value in this Task. If the current user is not a Participant, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

4. The protocol server MUST send the Task object back to the protocol client.

3.1.4.5.1 Messages

3.1.4.5.1.1 GetTaskInfoSoapIn

The request message sent by the protocol client to call the GetTaskInfo method on the protocol server.

The SOAP Action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskInfo
```

The SOAP Body contains a GetTaskInfo element.

3.1.4.5.1.2 GetTaskInfoSoapOut

The response sent by the protocol server upon the successful completion of the GetTaskInfo call.

The SOAP Action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskInfo
```

The SOAP Body contains a GetTaskInfoResponse element.

3.1.4.5.2 Elements

3.1.4.5.2.1 GetTaskInfo

This element specifies the name and the parameters of the GetTaskInfo Web service method.

```
<s:element name="GetTaskInfo">
  <s:complexType>
    <s:sequence>
      <s:element name="taskID" type="tns:guid"/>
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

taskID: A GUIDString that identifies the Task that the protocol client is requesting.

actingUser: When using trusted authentication, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.5.2.2 GetTaskInfoResponse

This element specifies the result of a successful completion of the GetTaskInfo Web service call.

```
<s:element name="GetTaskInfoResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="Task" type="tns:Task" />
    </s:sequence>
  </s:complexType>
</s:element>
```

Task: An instance of type Task. The value of this element MUST NOT be NULL.

3.1.4.5.3 Complex Types

None.

3.1.4.5.4 Simple Types

None.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.4.6 GetTaskMessage

This operation is used to retrieve the Task Message for a Task.

```
<wsdl:operation name="GetTaskMessage">
    <wsdl:input message="GetTaskMessageSoapIn" />
    <wsdl:output message="GetTaskMessageSoapOut" />
</wsdl:operation>
```

The protocol client sends a GetTaskMessageSoapIn request message and the protocol server responds with a GetTaskMessageSoapOut response message, as follows:

1. If taskID is NULL GUID, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. The protocol server retrieves the Task Message for Task identified by taskID from the Tracking System. The current user MUST be either the Initiator or the Target of the Task. If the protocol server fails to retrieve the Task Message, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Retrieve".

3. The protocol server MUST send the Task Message back to the protocol client.

3.1.4.6.1 Messages

3.1.4.6.1.1 GetTaskMessageSoapIn

The request message sent by the protocol client to call the GetTaskMessage web method.

The SOAP action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskMessage
```

The SOAP body contains a GetTaskMessage element.

3.1.4.6.1.2 GetTaskMessageSoapOut

The response message sent by the protocol server upon successful completion of the GetTaskMessage web method.

The SOAP Action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskMessage
```

The SOAP Body contains a GetTaskMessageResponse element.

3.1.4.6.2 Elements

3.1.4.6.2.1 GetTaskMessage

This element specifies the name and the parameters for the GetTaskMessage web method.

```
<s:element name="GetTaskMessage">
  <s:complexType>
    <s:sequence>
      <s:element name="taskID" type="tns:guid"/>
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

taskID: A GUIDString that identifies the [Task](#) whose Task Message the protocol client is requesting.

actingUser: When using Trusted Authentication, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.6.2.2 GetTaskMessageResponse

This element specifies the result of the GetTaskMessage web method call.

```
<s:element name="GetTaskMessageResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="TaskMessage" type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

TaskMessage: The Task Message the protocol client is requesting. The value of this element MUST NOT be NULL.

3.1.4.6.3 Complex Types

None.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6 Groups

None.

3.1.4.6.7 Attribute Groups

None.

3.1.4.7 SendTaskResponse

This operation is used to send a **task response message** to a Task.

```
<wsdl:operation name="SendTaskResponse">
    <wsdl:input message="SendTaskResponseSoapIn" />
    <wsdl:output message="SendTaskResponseSoapOut" />
</wsdl:operation>
```

The protocol client sends a SendTaskResponseSoapIn request message and the protocol server responds with a SendTaskResponseSoapOut response message, as follows:

1. The protocol server MUST ensure that taskResponse is a valid XML document and that it conforms to the Task Message schema as specified in section 7.2. If taskResponse is not a valid XML document, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Response". If taskResponse does not conform to the Task Message schema, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.InvalidParameter".

2. The protocol server MUST ensures that the current user is the Target of the Task the response is directed to. If the current user is not the Target, the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Client.Hws.AccessDenied".

3. The protocol server MUST respond with the Task Response Message to the Action Instance that created the Task.

4. The protocol server MUST ensure that at least one Action Instance in the Activity Flow has not finished before posting the Task Response Message, otherwise the protocol server MUST respond with a SOAP Fault message with SOAP Fault Code "Server.Hws.Engine".

3.1.4.7.1 Messages

3.1.4.7.1.1 SendTaskResponseSoapIn

The request message sent by the protocol client to call the SendTaskResponse method.

The SOAP Action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/SendTaskResponse
```

The SOAP Body contains a SendTaskResponse element.

3.1.4.7.1.2 SendTaskResponseSoapOut

The response message sent by the protocol server upon successful completion of the SendTaskResponse method.

The SOAP Action value of the message is defined as:

```
http://microsoft.com/Biztalk2004/Hws/Hwsservice/SendTaskResponse
```

The SOAP Body contains a SendTaskResponseResponse element.

3.1.4.7.2 Elements

3.1.4.7.2.1 SendTaskResponse

This element specifies the name and parameters for the SendTaskResponse Web service method.

```
<s:element name="SendTaskResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="taskResponse" type="s:string" />
      <s:element name="actingUser" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

taskResponse: A string representation of an XML document that conforms to the Task Message schema as specified in section 7.2.

actingUser: When using trusted authentication, the name of the user who is participating in the Activity Flow. The value of this element MUST be NULL and MUST be ignored.

3.1.4.7.2.2 SendTaskResponseResponse

The element specifies the result of the SendTaskResponse web method call.

```
<s:element name="SendTaskResponseResponse">
  <s:complexType/>
</s:element>
```

3.1.4.7.3 Complex Types

None.

3.1.4.7.4 Simple Types

None.

3.1.4.7.5 Attributes

None.

3.1.4.7.6 Groups

None.

3.1.4.7.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

3.2 Client Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

3.2.1 Abstract Data Model

None.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Message Processing Events and Sequencing Rules

None.

3.2.5 Timer Events

None.

3.2.6 Other Local Events

None.

4   Protocol Examples

The following examples show sample interactions between protocol client and protocol server.

4.1   Returning the Activity List

The protocol server receives a request from a protocol client for the Actions available to the protocol client. The following example shows a request from a protocol client to GetActivityList. In the following example, note that the ActivityFlowID is not NULL GUID, is set to 7B4C7544-E672-4E28-A3E3-9C7537BF5721, and is part of the original request to the protocol server.

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityList"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 755
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:GetActivityList xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <tns:activityFlowID>7B4C7544-E672-4E28-A3E3-9C7537BF5721</tns:activityFlowID>
      <tns:parentActionInstanceID>00000000-0000-0000-0000-
000000000000</tns:parentActionInstanceID>
      <tns:parentTaskID>00000000-0000-0000-0000-000000000000</tns:parentTaskID>
      <tns:target>
</tns:target>
      <tns:actingUser>
</tns:actingUser>
    </tns:GetActivityList>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with an array of available actions. In the following example a single ExecutableAction is returned. There could be more than one action (1) sent to the protocol client (see section 3.1.4.4.1.2). Also note in the following example that the ExecutableAction has ActionTypeID acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266, is of type Action, and is named Approval.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:13:48 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 843

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetActivityListResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <ExecutableActions>
        <ActivityType>Action</ActivityType>
        <ActionTypeID>acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266</ActionTypeID>
        <ActivityModelTypeID>00000000-0000-0000-0000-000000000000</ActivityModelTypeID>
        <ActivationBlockID>00000000-0000-0000-0000-000000000000</ActivationBlockID>
        <ActionIndex>0</ActionIndex>
        <PreviousActionTypeID>00000000-0000-0000-0000-000000000000</PreviousActionTypeID>
        <IsDependentComposible>false</IsDependentComposible>
        <Name>Approval</Name>
        <Description />
      </ExecutableActions>
    </GetActivityListResponse>
  </soap:Body>
</soap:Envelope>
```

4.2 Starting the Activity Flow

The protocol server starts the Workflow process after receiving a request from the protocol client to AddActionToActivityFlow. In the following example, the protocol client provides the ActivityFlowID 586BB6B6-8F29-438D-9F31-56A2451B4F25 on which to add the action (1) of type ActionTypeID ACC7F447-9EE2-FDF9-FE8B-FB4B0E8F3266. The protocol client provides the encoded Activation xml in the parametersDoc. The Activation xml contains the InfoPath form instance in base64Binary format.

```
POST /hwsservice/hwsservice.asmx?op=%0a%20%20%20%20%20%20%20%20GetActivityList HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/AddActionToActivityFlow"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 7442
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <AddActionToActivityFlow xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <activityFlowID>586BB6B6-8F29-438D-9F31-56A2451B4F25</activityFlowID>
      <parentActionInstanceID>00000000-0000-0000-0000-000000000000</parentActionInstanceID>
      <parentTaskID>00000000-0000-0000-0000-000000000000</parentTaskID>
      <isDependentComposition>false</isDependentComposition>
      <actionParameters>
        <Action>
          <ActivityType>Action</ActivityType>
          <ActionTypeID>ACC7F447-9EE2-FDF9-FE8B-FB4B0E8F3266</ActionTypeID>
          <ActivityModelTypeID>00000000-0000-0000-0000-000000000000</ActivityModelTypeID>
          <ActivationBlockID>00000000-0000-0000-0000-000000000000</ActivationBlockID>
          <ActionIndex>0</ActionIndex>
          <PreviousActionTypeID>00000000-0000-0000-0000-000000000000</PreviousActionTypeID>
```

```
                <IsDependentComposible>false</IsDependentComposible>
                <Name />
            </Action>
            <XsdDoc />
            <TaskSchemas />
            <Targets />
            <ParametersDoc>
                &lt;ns0:HwsMessage
 xmlns:ns0=&quot;http://tempuri.org/Hws_Activate_Approval&quot;&gt;
                &lt;HwsSection HwsMessageType=&quot;Hws_Activate&quot;&gt;
                &lt;ActivityFlowID&gt;ActivityFlowID_0&lt;/ActivityFlowID&gt;

 &lt;ActivityFlowDescription&gt;ActivityFlowDescription_0&lt;/ActivityFlowDescription&gt;
                &lt;InitiatingActor&gt;InitiatingActor_0&lt;/InitiatingActor&gt;
                &lt;ActionTypeID&gt;ActionTypeID_0&lt;/ActionTypeID&gt;
                &lt;ActionInstanceID&gt;ActionInstanceID_0&lt;/ActionInstanceID&gt;

 &lt;ActionInstanceDescription&gt;ActionInstanceDescription_0&lt;/ActionInstanceDescription&gt
 ;

 &lt;ParentActionInstanceID&gt;ParentActionInstanceID_0&lt;/ParentActionInstanceID&gt;
                &lt;ParentTaskID&gt;ParentTaskID_0&lt;/ParentTaskID&gt;
                &lt;ActivityModelTypeID&gt;ActivityModelTypeID_0&lt;/ActivityModelTypeID&gt;

 &lt;ActivityModelInstanceID&gt;ActivityModelInstanceID_0&lt;/ActivityModelInstanceID&gt;
                &lt;ActivityModelStepID&gt;0&lt;/ActivityModelStepID&gt;
                &lt;IsDependentOnParent&gt;false&lt;/IsDependentOnParent&gt;
                &lt;ActivityFlowProperties&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_0&lt;/Property&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_1&lt;/Property&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_2&lt;/Property&gt;
                &lt;/ActivityFlowProperties&gt;
                &lt;ActionProperties&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_0&lt;/Property&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_1&lt;/Property&gt;
                &lt;Property Name=&quot;Name_0&quot; Description=&quot;Description_1&quot;
 Type=&quot;Type_2&quot;&gt;Property_2&lt;/Property&gt;
                &lt;/ActionProperties&gt;
                &lt;HwsWebServiceUrl&gt;HwsWebServiceUrl_0&lt;/HwsWebServiceUrl&gt;
                &lt;/HwsSection&gt;
                &lt;ActionSection&gt;
                &lt;Target&gt;domain\taskTarget&lt;/Target&gt;
                &lt;Description/&gt;
                &lt;Status&gt;Accepted&lt;/Status&gt;
                &lt;EndAction&gt;false&lt;/EndAction&gt;
                &lt;/ActionSection&gt;
                &lt;Payloads&gt;
                &lt;Payload ID=&quot;ID_0&quot;&gt;
                &lt;/Payload&gt;
                &lt;Payload ID=&quot;IP2003&quot;&gt;
 PD94bWwgdmVyc2lvbj0iMS4wIj8+PD9tc28taW5mb1BhdGhTb2x1dGlvbiBzb2x1dGlvblZlcnNpb249
 IjEuMC4wLjQiIHByb2R1Y3RWZXJzaW9uPSIxMi4wLjAiIFBJVmVyc2lvbj0iMS4wLjAuMCIgaHJlZj0i
 ZmlsZTovLy9DOlxIV1NJbmZvUGF0aFNvbHV0aW9uXG1hbmlmZXN0LnhzZiIgbmFtZT0idXJuOnNjaG…&lt;/Payload&g
 t;&lt;/Payloads&gt;
                &lt;/ns0:HwsMessage&gt;
```

```
            </ParametersDoc>
            <ActionInstanceTitle>Approval</ActionInstanceTitle>
            <IsOverridable>false</IsOverridable>
          </actionParameters>
          <actingUser />
        </AddActionToActivityFlow>
      </SOAP-ENV:Body>
    </SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the ActionInstanceID b3fdc1eb-3338-4de4-a11c-6eed71393e44 that has been added to the Workflow.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:13:56 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 442

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<AddActionToActivityFlowResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
<ActionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</ActionInstanceID>
</AddActionToActivityFlowResponse>
</soap:Body>
</soap:Envelope>
```

The protocol server next provides Workflow status information by first receiving a request to GetActionInstance. The workflow client calls GetActionInstance and provides the identifier b3fdc1eb-3338-4de4-a11c-6eed71393e44 of the target action (1).

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActionInstance"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 572
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:GetActionInstance xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <tns:actionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</tns:actionInstanceID>
      <tns:actingUser>
</tns:actingUser>
    </tns:GetActionInstance>
```

```
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the Action Instance information. This includes any Tasks that are part of this Action Instance. The following example shows that Activity Flow 586bb6b6-8f29-438d-9f31-56a2451b4f25 has one Action Instance b3fdc1eb-3338-4de4-a11c-6eed71393e44 of type acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266 with a Task instance b749e695-a805-4624-afbe-8f128a5fa172 assigned to domain\taskTarget. The current Action Instance is in the Start state and the Task is in the Accepted state.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:00 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 3020

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetActionInstanceResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <ActionInstance>
        <ActivityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</ActivityFlowID>
        <ActionTypeID>acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266</ActionTypeID>
        <ActionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</ActionInstanceID>
        <PreviousActionInstanceID>00000000-0000-0000-0000-
000000000000</PreviousActionInstanceID>
        <PreviousTaskID>00000000-0000-0000-0000-000000000000</PreviousTaskID>
        <ActivityModelTypeID>00000000-0000-0000-0000-000000000000</ActivityModelTypeID>
        <ActivityModelIndex>0</ActivityModelIndex>
        <ActivityModelInstanceID>00000000-0000-0000-0000-
000000000000</ActivityModelInstanceID>
        <Name>Approval</Name>
        <ActionInstanceTitle>Approval</ActionInstanceTitle>
        <Initiator>domain\taskInitiator</Initiator>
        <IsDependent>false</IsDependent>
        <Status>Start</Status>
        <StartTime>2008-02-13T17:13:56.9770000-08:00</StartTime>
        <EndTime>2008-02-13T17:13:57.9670000-08:00</EndTime>
        <Tasks>
          <Task>
            <Initiator>domain\taskInitiator</Initiator>
            <Target>domain\taskTarget</Target>
            <TargetXPath />
            <TaskID>b749e695-a805-4624-afbe-8f128a5fa172</TaskID>
            <TaskDescription />
            <ActivityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</ActivityFlowID>
            <ActionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</ActionInstanceID>
            <TaskTimeStamp>2008-02-13T17:13:57.9670000-08:00</TaskTimeStamp>
            <CurrentStatus>Accepted</CurrentStatus>
            <CurrentPercentageComplete>100</CurrentPercentageComplete>
            <TargetNamespaceUri>http://tempuri.org/Hws_Task_Approval</TargetNamespaceUri>
```

```xml
            <TaskProperties>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_0</Value>
              </ResourceProperty>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_1</Value>
              </ResourceProperty>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_2</Value>
              </ResourceProperty>
            </TaskProperties>
            <Reassignment>
              <PreviousTaskID>00000000-0000-0000-0000-000000000000</PreviousTaskID>
              <NextTaskID>00000000-0000-0000-0000-000000000000</NextTaskID>
              <FirstTaskID>00000000-0000-0000-0000-000000000000</FirstTaskID>
              <LastTaskID>00000000-0000-0000-0000-000000000000</LastTaskID>
            </Reassignment>
          </Task>
        </Tasks>
        <ActionProperties>
          <ResourceProperty>
            <Name>Name_0</Name>
            <Type>Type_2</Type>
            <Description>Description_1</Description>
            <Value>Property_0</Value>
          </ResourceProperty>
          <ResourceProperty>
            <Name>Name_0</Name>
            <Type>Type_2</Type>
            <Description>Description_1</Description>
            <Value>Property_1</Value>
          </ResourceProperty>
          <ResourceProperty>
            <Name>Name_0</Name>
            <Type>Type_2</Type>
            <Description>Description_1</Description>
            <Value>Property_2</Value>
          </ResourceProperty>
        </ActionProperties>
      </ActionInstance>
    </GetActionInstanceResponse>
  </soap:Body>
</soap:Envelope>
```

4.3 Getting the Task Status

The protocol server provides workflow Task status information by first receiving a request to GetTaskInfo. In the following example, the protocol client calls GetTaskInfo and provides the TaskID b749e695-a805-4624-afbe-8f128a5fa172.

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskInfo"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 540
Pragma: no-cache
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:GetTaskInfo xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <tns:taskID>b749e695-a805-4624-afbe-8f128a5fa172</tns:taskID>
      <tns:actingUser>
</tns:actingUser>
    </tns:GetTaskInfo>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the Task instance information. The following example shows the Task information for Task b749e695-a805-4624-afbe-8f128a5fa172. The following task is in the Accepted state and is assigned to domain\taskTarget.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:07 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 1658

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetTaskInfoResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <Task>
        <Initiator>domain\taskInitiator</Initiator>
        <Target>domain\taskTarget</Target>
        <TargetXPath />
        <TaskID>b749e695-a805-4624-afbe-8f128a5fa172</TaskID>
        <TaskDescription />
        <ActivityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</ActivityFlowID>
        <ActionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</ActionInstanceID>
        <TaskTimeStamp>2008-02-13T17:13:57.9670000-08:00</TaskTimeStamp>
        <CurrentStatus>Accepted</CurrentStatus>
        <CurrentPercentageComplete>100</CurrentPercentageComplete>
```

```
            <TargetNamespaceUri>http://tempuri.org/Hws_Task_Approval</TargetNamespaceUri>
            <TaskProperties>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_0</Value>
              </ResourceProperty>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_1</Value>
              </ResourceProperty>
              <ResourceProperty>
                <Name>Name_0</Name>
                <Type>Type_2</Type>
                <Description>Description_1</Description>
                <Value>Property_2</Value>
              </ResourceProperty>
            </TaskProperties>
            <Reassignment>
              <PreviousTaskID>00000000-0000-0000-0000-000000000000</PreviousTaskID>
              <NextTaskID>00000000-0000-0000-0000-000000000000</NextTaskID>
              <FirstTaskID>00000000-0000-0000-0000-000000000000</FirstTaskID>
              <LastTaskID>00000000-0000-0000-0000-000000000000</LastTaskID>
            </Reassignment>
          </Task>
        </GetTaskInfoResponse>
      </soap:Body>
    </soap:Envelope>
```

4.3.1   Getting the Activity Flow Information

The protocol server provides Activity Flow information to the protocol client by first receiving a request to GetActivityFlowInfo and passing in the Activity Flow identifier 586bb6b6-8f29-438d-9f31-56a2451b4f25. The request also contains the DetailLevel set to ActionInstanceLevel.

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityFlowInfo"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 626
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:GetActivityFlowInfo xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <tns:activityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</tns:activityFlowID>
      <tns:detailLevel>ActionInstanceLevel</tns:detailLevel>
      <tns:actingUser>
</tns:actingUser>
```

```
        </tns:GetActivityFlowInfo>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the Activity Flow information. The following example shows the Activity Flow information for Activity Flow 586bb6b6-8f29-438d-9f31-56a2451b4f25.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:07 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 3018

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetActivityFlowInfoResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <ActivityFlowTrackingInfo>
        <DetailLevel>ActionInstanceLevel</DetailLevel>
        <StatInfo>
          <DetailLevel>ActionInstanceLevel</DetailLevel>
          <ActionInstanceCount>1</ActionInstanceCount>
          <AdHocActionInstanceCount>1</AdHocActionInstanceCount>
          <ActivityModelCount>0</ActivityModelCount>
          <LastActionName>Approval</LastActionName>
          <LastActionInitiator>doomain\taskInitiator</LastActionInitiator>
          <LastActionTargets />
        </StatInfo>
        <ActivityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</ActivityFlowID>
        <ActivityFlowDescription>ActivityFlowDescription_0</ActivityFlowDescription>
        <Status>Active</Status>
        <RootActionInstances>
          <ActionInstance>
            <ActivityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</ActivityFlowID>
            <ActionTypeID>acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266</ActionTypeID>
            <ActionInstanceID>b3fdc1eb-3338-4de4-a11c-6eed71393e44</ActionInstanceID>
            <PreviousActionInstanceID>00000000-0000-0000-0000-
000000000000</PreviousActionInstanceID>
            <PreviousTaskID>00000000-0000-0000-0000-000000000000</PreviousTaskID>
            <ActivityModelTypeID>00000000-0000-0000-0000-000000000000</ActivityModelTypeID>
            <ActivityModelIndex>0</ActivityModelIndex>
            <ActivityModelInstanceID>00000000-0000-0000-0000-
000000000000</ActivityModelInstanceID>
            <Name>Approval</Name>
            <ActionInstanceTitle>Approval</ActionInstanceTitle>
            <Initiator>domain\taskInitiator</Initiator>
            <IsDependent>false</IsDependent>
            <Status>Start</Status>
            <StartTime>2008-02-13T17:13:56.9770000-08:00</StartTime>
            <EndTime>2008-02-13T17:13:57.9670000-08:00</EndTime>
            <ActionProperties>
```

```
                    <ResourceProperty>
                      <Name>Name_0</Name>
                      <Type>Type_2</Type>
                      <Description>Description_1</Description>
                      <Value>Property_0</Value>
                    </ResourceProperty>
                    <ResourceProperty>
                      <Name>Name_0</Name>
                      <Type>Type_2</Type>
                      <Description>Description_1</Description>
                      <Value>Property_1</Value>
                    </ResourceProperty>
                    <ResourceProperty>
                      <Name>Name_0</Name>
                      <Type>Type_2</Type>
                      <Description>Description_1</Description>
                      <Value>Property_2</Value>
                    </ResourceProperty>
                  </ActionProperties>
                </ActionInstance>
              </RootActionInstances>
              <Initiator>domain\taskInitiator</Initiator>
              <AllIgnoring>false</AllIgnoring>
              <StartTimeStamp>2008-02-13T17:13:56.9770000-08:00</StartTimeStamp>
              <LastModifiedTimeStamp>2008-02-13T17:13:57.9670000-08:00</LastModifiedTimeStamp>
              <ActivityFlowProperties>
                <ResourceProperty>
                  <Name>Name_0</Name>
                  <Type>Type_2</Type>
                  <Description>Description_1</Description>
                  <Value>Property_0</Value>
                </ResourceProperty>
                <ResourceProperty>
                  <Name>Name_0</Name>
                  <Type>Type_2</Type>
                  <Description>Description_1</Description>
                  <Value>Property_1</Value>
                </ResourceProperty>
                <ResourceProperty>
                  <Name>Name_0</Name>
                  <Type>Type_2</Type>
                  <Description>Description_1</Description>
                  <Value>Property_2</Value>
                </ResourceProperty>
              </ActivityFlowProperties>
            </ActivityFlowTrackingInfo>
          </GetActivityFlowInfoResponse>
        </soap:Body>
      </soap:Envelope>
```

4.3.2 Getting the Activity List for a Running Workflow

The protocol server receives a request from a protocol client for the available Actions available to the protocol client. The following example shows a request from a protocol client to the GetActivityList as specified in section 3.1.4.4. Note that the Activity Flow identifier is 586bb6b6-8f29-438d-9f31-56a2451b4f25, the parent action instance identifier is b3fdc1eb-3338-4de4-a11c-6eed71393e44 and the parent task identifier is b749e695-a805-4624-afbe-8f128a5fa172.

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityList"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 755
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:GetActivityList xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <tns:activityFlowID>586bb6b6-8f29-438d-9f31-56a2451b4f25</tns:activityFlowID>
      <tns:parentActionInstanceID>b3fdc1eb-3338-4de4-a11c-
6eed71393e44</tns:parentActionInstanceID>
      <tns:parentTaskID>b749e695-a805-4624-afbe-8f128a5fa172</tns:parentTaskID>
      <tns:target>
</tns:target>
      <tns:actingUser>
</tns:actingUser>
    </tns:GetActivityList>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the available Actions. Note that the following example returns a single ExecutableAction. There could be more than one action (1) sent to the protocol client.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:07 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 843

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetActivityListResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <ExecutableActions>
        <ActivityType>Action</ActivityType>
        <ActionTypeID>acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266</ActionTypeID>
        <ActivityModelTypeID>00000000-0000-0000-0000-000000000000</ActivityModelTypeID>
        <ActivationBlockID>00000000-0000-0000-0000-000000000000</ActivationBlockID>
        <ActionIndex>0</ActionIndex>
        <PreviousActionTypeID>acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266</PreviousActionTypeID>
        <IsDependentComposible>false</IsDependentComposible>
        <Name>Approval</Name>
        <Description />
```

```
          </ExecutableActions>
        </GetActivityListResponse>
      </soap:Body>
    </soap:Envelope>
```

4.4   Providing the Task Message

The protocol server provides the protocol client with the Task Message by first receiving a request to
GetTaskMessage. The protocol client provides the Task identifier b749e695-a805-4624-afbe-
8f128a5fa172.

4.4.1   GetTaskMessage

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskMessage"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 513
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetTaskMessage xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <taskID>b749e695-a805-4624-afbe-8f128a5fa172</taskID>
      <actingUser />
    </GetTaskMessage>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The protocol server responds to the protocol client with the Task Message.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:13 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 5803

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetTaskMessageResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <TaskMessage>
        &lt;ns0:HwsMessage xmlns:ns0="http://tempuri.org/Hws_Task_Approval"&gt;&lt;HwsSection
HwsMessageType="Hws_Task"&gt;&lt;ActivityFlowID&gt;586bb6b6-8f29-438d-9f31-
56a2451b4f25&lt;/ActivityFlowID&gt;&lt;TaskID&gt;b749e695-a805-4624-afbe-
```

```
8f128a5fa172&lt;/TaskID&gt;&lt;TaskDescription&gt;&lt;/TaskDescription&gt;&lt;ActionTypeID&gt
;acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266&lt;/ActionTypeID&gt;&lt;ActionInstanceID&gt;b3fdc1eb-
3338-4de4-a11c-
6eed71393e44&lt;/ActionInstanceID&gt;&lt;InitiatingActor&gt;DOMAIN\taskInitiator&lt;/Initiati
ngActor&gt;&lt;ActorElementXPath&gt;DOMAIN\taskInitiator&lt;/ActorElementXPath&gt;&lt;TargetA
ctor&gt;domain\taskTarget&lt;/TargetActor&gt;&lt;ActivityModelTypeID&gt;00000000-0000-0000-
0000-
000000000000&lt;/ActivityModelTypeID&gt;&lt;ActivityModelStepID&gt;0&lt;/ActivityModelStepID&
gt;&lt;TaskProperties&gt;&lt;Property Name="Name_0" Description="Description_1"
Type="Type_2"&gt;Property_0&lt;/Property&gt;&lt;Property Name="Name_0"
Description="Description_1" Type="Type_2"&gt;Property_1&lt;/Property&gt;&lt;Property
Name="Name_0" Description="Description_1"
Type="Type_2"&gt;Property_2&lt;/Property&gt;&lt;/TaskProperties&gt;&lt;ActivityModelInstanceI
D&gt;00000000-0000-0000-0000-
000000000000&lt;/ActivityModelInstanceID&gt;&lt;HwsWebServiceUrl&gt;http://hostServer/hwsserv
ice/hwsservice.asmx?op=

GetActivityList&lt;/HwsWebServiceUrl&gt;&lt;TaskStatus&gt;Accepted&lt;/TaskStatus&gt;&lt;Perc
entageComplete&gt;100&lt;/PercentageComplete&gt;&lt;NumberOfResponses&gt;0&lt;/NumberOfRespon
ses&gt;&lt;Reassignment&gt;&lt;ReassignedByActor&gt;actor&lt;/ReassignedByActor&gt;&lt;FromTa
skID&gt;00000000-0000-0000-0000-
000000000000&lt;/FromTaskID&gt;&lt;/Reassignment&gt;&lt;/HwsSection&gt;&lt;ActionSection&gt;&
lt;EndAction&gt;false&lt;/EndAction&gt;&lt;/ActionSection&gt;&lt;Payloads&gt;&lt;Payload
ID="ID_0"&gt;&lt;/Payload&gt;&lt;Payload ID="IP2003"&gt;
PD94bWwgdmVyc2lvbj0iMS4wIj8+PD9tc28taW5mb1BhdGhTb2x1dGlvbiBzb2x1dGlvblZlcnNpb249
IjEuMC4wLjQiIHByb2R1Y3RWZXJzaW9uPSIxMi4wIiwLjAiIFBJVmVyc2lvbj0iMS4wLjAuMCIgaHJlZj0i
ZmlsZTovLy9DOlxIV1NjbmZvUGF0aFNvbHV0aW9uXG1hbmlmZXN0Lnhzbii IgbmFtZT0idXJuOnN…&lt;/Payload&gt;&
lt;/Payloads&gt;&lt;/ns0:HwsMessage&gt;
        &lt;/TaskMessage&gt;
      &lt;/GetTaskMessageResponse&gt;
   &lt;/soap:Body&gt;
&lt;/soap:Envelope&gt;
```

4.4.2 Sending the Task Response

The protocol server responds to a Task Response by first receiving a call from the protocol client to
SendTaskResponse. The protocol client provides the Task Message within the SendTaskResponse
call.

```
POST /hwsservice/hwsservice.asmx HTTP/1.1
SOAPAction: "http://microsoft.com/Biztalk2004/Hws/Hwsservice/SendTaskResponse"
Content-Type: text/xml; charset="UTF-8"
User-Agent: SOAP Toolkit 3.0
Host: hostServer
Content-Length: 6534
Pragma: no-cache

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SendTaskResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <taskResponse>
        &lt;ns0:HwsMessage
xmlns:ns0=&quot;http://tempuri.org/Hws_Task_Approval&quot;&gt;&lt;HwsSection
HwsMessageType=&quot;Hws_Task&quot;&gt;&lt;ActivityFlowID&gt;586bb6b6-8f29-438d-9f31-
56a2451b4f25&lt;/ActivityFlowID&gt;&lt;TaskID&gt;b749e695-a805-4624-afbe-
8f128a5fa172&lt;/TaskID&gt;&lt;TaskDescription&gt;&lt;/TaskDescription&gt;&lt;ActionTypeID&gt
;acc7f447-9ee2-fdf9-fe8b-fb4b0e8f3266&lt;/ActionTypeID&gt;&lt;ActionInstanceID&gt;b3fdc1eb-
```

```
3338-4de4-a11c-
6eed71393e44&lt;/ActionInstanceID&gt;&lt;InitiatingActor&gt;DOMAIN\taskInitiator&lt;/Initiati
ngActor&gt;&lt;ActorElementXPath&gt;DOMAIN\taskInitiator&lt;/ActorElementXPath&gt;&lt;TargetA
ctor&gt;domain\taskTarget&lt;/TargetActor&gt;&lt;ActivityModelTypeID&gt;00000000-0000-0000-
0000-
000000000000&lt;/ActivityModelTypeID&gt;&lt;ActivityModelStepID&gt;0&lt;/ActivityModelStepID&
gt;&lt;TaskProperties&gt;&lt;Property Name=&quot;Name_0&quot;
Description=&quot;Description_1&quot;
Type=&quot;Type_2&quot;&gt;Property_0&lt;/Property&gt;&lt;Property Name=&quot;Name_0&quot;
Description=&quot;Description_1&quot;
Type=&quot;Type_2&quot;&gt;Property_1&lt;/Property&gt;&lt;Property Name=&quot;Name_0&quot;
Description=&quot;Description_1&quot;
Type=&quot;Type_2&quot;&gt;Property_2&lt;/Property&gt;&lt;/TaskProperties&gt;&lt;ActivityMode
lInstanceID&gt;00000000-0000-0000-0000-
000000000000&lt;/ActivityModelInstanceID&gt;&lt;HwsWebServiceUrl&gt;http://hostServer/hwsserv
ice/hwsservice.asmx?op=

GetActivityList&lt;/HwsWebServiceUrl&gt;&lt;TaskStatus&gt;Accepted&lt;/TaskStatus&gt;&lt;Perc
entageComplete&gt;100&lt;/PercentageComplete&gt;&lt;NumberOfResponses&gt;0&lt;/NumberOfRespon
ses&gt;&lt;Reassignment&gt;&lt;ReassignedByActor&gt;actor&lt;/ReassignedByActor&gt;&lt;FromTa
skID&gt;00000000-0000-0000-0000-
000000000000&lt;/FromTaskID&gt;&lt;/Reassignment&gt;&lt;/HwsSection&gt;&lt;ActionSection&gt;&
lt;EndAction&gt;true&lt;/EndAction&gt;&lt;/ActionSection&gt;&lt;Payloads&gt;&lt;Payload
ID=&quot;ID_0&quot;&gt;&lt;/Payload&gt;&lt;Payload ID=&quot;IP2003&quot;&gt;
PD94bWwgdmVyc2lvbj0iMS4wIj8+PD9tc28taW5mb1BhdGhTb2x1dGlvbiBzb2x1dGlvblZlcnNpb249
IjEuMC4wLjQiIHByb2R1Y3RRWZXJzaW9uPSIxMi4wLjAiIFBBJVmVyc2lvbj0iMS4wLjAuMCIgaHJlZj0i
ZmlsZTovLy9DOlxIV1NJbmZvUGF0aFNvbHV0aW9uXG1hbmlmZXN0LnhzaiIgbmFt…&lt;/Payload&gt;&lt;/Payload
s&gt;&lt;/ns0:HwsMessage&gt;
        </taskResponse>
        <actingUser />
      </SendTaskResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The protocol server processes the Task Message and responds to the protocol client.

```
HTTP/1.1 200 OK
Date: Wed, 13 Feb 2008 17:14:14 GMT
Server: Microsoft-IIS/6.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: private, max-age=0
Content-Type: text/xml; charset=utf-8
Content-Length: 330

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<SendTaskResponseResponse xmlns="http://microsoft.com/Biztalk2004/Hws/Hwsservice" />
</soap:Body>
</soap:Envelope>
```

5   Security

5.1   Security Considerations for Implementers

In addition to the security considerations applicable to the underlying protocols, it is advised that the protocol server require that all requests come from authenticated users only.

5.2   Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation the full Web Services Description Language (WSDL) follows:

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://microsoft.com/Biztalk2004/Hws/Hwsservice"
targetNamespace="http://microsoft.com/Biztalk2004/Hws/Hwsservice"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified"
targetNamespace="http://microsoft.com/Biztalk2004/Hws/Hwsservice">
      <s:import namespace="http://microsoft.com/wsdl/types/" />
      <s:element name="GetActivityList">
        <s:complexType>
          <s:sequence>
            <s:element name="activityFlowID" type="tns:guid" />
            <s:element name="parentActionInstanceID" type="tns:guid" />
            <s:element name="parentTaskID" type="tns:guid" />
            <s:element minOccurs="0" maxOccurs="1" name="target" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:simpleType name="guid">
        <s:restriction base="s:string">
          <s:pattern value="[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-
9a-fA-F]{12}"/>
        </s:restriction>
      </s:simpleType>
      <s:element name="GetActivityListResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="ExecutableActions"
type="tns:Activity" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="Activity">
        <s:sequence>
          <s:element name="ActivityType" type="tns:ActivityType" />
          <s:element name="ActionTypeID" type="tns:guid" />
          <s:element name="ActivityModelTypeID" type="tns:guid" />
          <s:element name="ActivationBlockID" type="tns:guid" />
          <s:element name="ActionIndex" type="s:int" />
          <s:element minOccurs="0" maxOccurs="1" name="StepName" type="s:string" />
          <s:element minOccurs="0" maxOccurs="1" name="StepDescription" type="s:string" />
          <s:element name="PreviousActionTypeID" type="tns:guid" />
          <s:element name="IsDependentComposible" type="s:boolean" />
          <s:element name="Name" type="s:string" />
          <s:element name="Description" type="s:string" />
        </s:sequence>
      </s:complexType>
      <s:simpleType name="ActivityType">
        <s:restriction base="s:string">
          <s:enumeration value="Action" />
          <s:enumeration value="ActivityModel" />
        </s:restriction>
```

```
        </s:simpleType>
        <s:element name="AddActionToActivityFlow">
          <s:complexType>
            <s:sequence>
              <s:element name="activityFlowID" type="tns:guid" />
              <s:element name="parentActionInstanceID" type="tns:guid" />
              <s:element name="parentTaskID" type="tns:guid" />
              <s:element name="isDependentComposition" type="s:boolean" />
              <s:element name="actionParameters" type="tns:ActionParameters" />
              <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:complexType name="ActionParameters">
          <s:sequence>
            <s:element name="Action" type="tns:Activity" />
            <s:element minOccurs="0" maxOccurs="1" name="XsdDoc" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="TaskSchemas" type="tns:ArrayOfString"
  />
            <s:element minOccurs="0" maxOccurs="1" name="Targets" type="tns:ArrayOfTarget" />
            <s:element name="ParametersDoc" type="s:string" />
            <s:element name="ActionInstanceTitle" type="s:string" />
            <s:element name="IsOverridable" type="s:boolean" />
          </s:sequence>
        </s:complexType>
        <s:complexType name="ArrayOfString">
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
  type="s:string" />
          </s:sequence>
        </s:complexType>
        <s:complexType name="ArrayOfTarget">
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="Target" nillable="true"
  type="tns:Target" />
          </s:sequence>
        </s:complexType>
        <s:complexType name="Target">
          <s:sequence>
            <s:element name="ParameterName" type="s:string" />
            <s:element name="InstanceXPath" type="s:string" />
            <s:element name="Min" type="s:int" />
            <s:element name="Max" type="s:int" />
          </s:sequence>
        </s:complexType>
        <s:element name="AddActionToActivityFlowResponse">
          <s:complexType>
            <s:sequence>
              <s:element name="ActionInstanceID" type="tns:guid" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetActivityFlowInfo">
          <s:complexType>
            <s:sequence>
              <s:element name="activityFlowID" type="tns:guid" />
              <s:element name="detailLevel" type="tns:ActivityFlowDetailLevel" />
              <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
            </s:sequence>
```

```xml
        </s:complexType>
      </s:element>
      <s:simpleType name="ActivityFlowDetailLevel">
        <s:restriction base="s:string">
          <s:enumeration value="ActionInstanceLevel" />
        </s:restriction>
      </s:simpleType>
      <s:element name="GetActivityFlowInfoResponse">
        <s:complexType>
          <s:sequence>
            <s:element name="ActivityFlowTrackingInfo" type="tns:ActivityFlow" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="ActivityFlow">
        <s:sequence>
          <s:element name="DetailLevel" type="tns:ActivityFlowDetailLevel" />
          <s:element minOccurs="0" maxOccurs="1" name="StatInfo" type="tns:ActivityFlowStat"
 />
          <s:element name="ActivityFlowID" type="tns:guid" />
          <s:element minOccurs="0" maxOccurs="1" name="ActivityFlowDescription"
type="s:string" />
          <s:element name="Status" type="tns:ActivityFlowStatus" />
          <s:element name="RootActionInstances" type="tns:ArrayOfActionInstance" />
          <s:element name="Initiator" type="s:string" />
          <s:element name="AllIgnoring" type="s:boolean" />
          <s:element name="StartTimeStamp" type="s:dateTime" />
          <s:element name="LastModifiedTimeStamp" type="s:dateTime" />
          <s:element minOccurs="0" maxOccurs="1" name="ActivityFlowProperties"
type="tns:ArrayOfResourceProperty" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="ActivityFlowStat">
        <s:sequence>
          <s:element name="DetailLevel" type="tns:ActivityFlowDetailLevel" />
          <s:element name="ActionInstanceCount" type="s:int" />
          <s:element name="AdHocActionInstanceCount" type="s:int" />
          <s:element name="ActivityModelCount" type="s:int" />
          <s:element minOccurs="0" maxOccurs="1" name="LastActionName" type="s:string" />
          <s:element minOccurs="0" maxOccurs="1" name="LastActionInitiator" type="s:string"
 />
          <s:element minOccurs="0" maxOccurs="1" name="LastActionTargets"
type="tns:ArrayOfString" />
        </s:sequence>
      </s:complexType>
      <s:simpleType name="ActivityFlowStatus">
        <s:restriction base="s:string">
          <s:enumeration value="Active" />
          <s:enumeration value="InActive" />
          <s:enumeration value="Complete" />
        </s:restriction>
      </s:simpleType>
      <s:complexType name="ArrayOfActionInstance">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="ActionInstance"
nillable="true" type="tns:ActionInstance" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="ActionInstance">
```

```
        <s:sequence>
          <s:element name="ActivityFlowID" type="tns:guid" />
          <s:element name="ActionTypeID" type="tns:guid" />
          <s:element name="ActionInstanceID" type="tns:guid" />
          <s:element name="PreviousActionInstanceID" type="tns:guid" />
          <s:element name="PreviousTaskID" type="tns:guid" />
          <s:element name="ActivityModelTypeID" type="tns:guid" />
          <s:element name="ActivityModelIndex" type="s:int" />
          <s:element name="ActivityModelInstanceID" type="tns:guid" />
          <s:element name="Name" type="s:string" />
          <s:element minOccurs="0" maxOccurs="1" name="ActivityModelName" type="s:string" />
          <s:element name="ActionInstanceTitle" type="s:string" />
          <s:element name="Initiator" type="s:string" />
          <s:element name="IsDependent" type="s:boolean" />
          <s:element name="Status" type="tns:ActionStatus" />
          <s:element name="StartTime" type="s:dateTime" />
          <s:element name="EndTime" type="s:dateTime" />
          <s:element minOccurs="0" maxOccurs="1" name="Tasks" type="tns:ArrayOfTask" />
          <s:element minOccurs="0" maxOccurs="1" name="ActionProperties"
  type="tns:ArrayOfResourceProperty" />
          <s:element minOccurs="0" maxOccurs="1" name="ChildActionInstances"
  type="tns:ArrayOfActionInstance" />
        </s:sequence>
      </s:complexType>
      <s:simpleType name="ActionStatus">
        <s:restriction base="s:string">
          <s:enumeration value="Start" />
          <s:enumeration value="Finish" />
          <s:enumeration value="Abort" />
        </s:restriction>
      </s:simpleType>
      <s:complexType name="ArrayOfTask">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="Task" nillable="true"
  type="tns:Task" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="Task">
        <s:sequence>
          <s:element name="Initiator" type="s:string" />
          <s:element name="Target" type="s:string" />
          <s:element minOccurs="0" maxOccurs="1" name="TargetXPath" type="s:string" />
          <s:element name="TaskID" type="tns:guid" />
          <s:element minOccurs="0" maxOccurs="1" name="TaskDescription" type="s:string" />
          <s:element name="ActivityFlowID" type="tns:guid" />
          <s:element name="ActionInstanceID" type="tns:guid" />
          <s:element name="TaskTimeStamp" type="s:dateTime" />
          <s:element name="CurrentStatus" type="tns:TaskStatus" />
          <s:element name="CurrentPercentageComplete" type="s:int" />
          <s:element minOccurs="0" maxOccurs="1" name="Responses"
  type="tns:ArrayOfTaskResponse" />
          <s:element name="TargetNamespaceUri" type="s:string" />
          <s:element minOccurs="0" maxOccurs="1" name="TaskProperties"
  type="tns:ArrayOfResourceProperty" />
          <s:element name="Reassignment" type="tns:Reassignment" />
        </s:sequence>
      </s:complexType>
      <s:simpleType name="TaskStatus">
        <s:restriction base="s:string">
```

```
                  <s:enumeration value="NotStarted" />
                  <s:enumeration value="InProgress" />
                  <s:enumeration value="Completed" />
                  <s:enumeration value="Waiting" />
                  <s:enumeration value="Deferred" />
                  <s:enumeration value="Accepted" />
                  <s:enumeration value="Tentative" />
                  <s:enumeration value="Declined" />
                  <s:enumeration value="Interrupted" />
                  <s:enumeration value="Cancelled" />
               </s:restriction>
            </s:simpleType>
            <s:complexType name="ArrayOfTaskResponse">
               <s:sequence>
                  <s:element minOccurs="0" maxOccurs="unbounded" name="TaskResponse" nillable="true"
    type="tns:TaskResponse" />
               </s:sequence>
            </s:complexType>
            <s:complexType name="TaskResponse">
               <s:sequence>
                  <s:element name="ResponseTimeStamp" type="s:dateTime" />
                  <s:element name="Status" type="tns:TaskStatus" />
                  <s:element name="PercentageComplete" type="s:int" />
                  <s:element name="Responder" type="s:string" />
                  <s:element minOccurs="0" maxOccurs="1" name="ResponseProperties"
    type="tns:ArrayOfResourceProperty" />
               </s:sequence>
            </s:complexType>
            <s:complexType name="ArrayOfResourceProperty">
               <s:sequence>
                  <s:element minOccurs="0" maxOccurs="unbounded" name="ResourceProperty"
    type="tns:ResourceProperty" />
               </s:sequence>
            </s:complexType>
            <s:complexType name="ResourceProperty">
               <s:sequence>
                  <s:element name="Name" type="s:string" />
                  <s:element name="Type" type="s:string" />
                  <s:element minOccurs="0" maxOccurs="1" name="Description" type="s:string" />
                  <s:element name="Value" type="s:string" />
               </s:sequence>
            </s:complexType>
            <s:complexType name="Reassignment">
               <s:sequence>
                  <s:element name="PreviousTaskID" type="tns:guid" />
                  <s:element name="NextTaskID" type="tns:guid" />
                  <s:element name="FirstTaskID" type="tns:guid" />
                  <s:element name="LastTaskID" type="tns:guid" />
               </s:sequence>
            </s:complexType>
            <s:element name="GetActionInstance">
               <s:complexType>
                  <s:sequence>
                     <s:element name="actionInstanceID" type="tns:guid" />
                     <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
                  </s:sequence>
               </s:complexType>
            </s:element>
            <s:element name="GetActionInstanceResponse">
```

```xml
          <s:complexType>
            <s:sequence>
              <s:element name="ActionInstance" type="tns:ActionInstance" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetTaskInfo">
          <s:complexType>
            <s:sequence>
              <s:element name="taskID" type="tns:guid" />
              <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetTaskInfoResponse">
          <s:complexType>
            <s:sequence>
              <s:element name="Task" type="tns:Task" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetTaskMessage">
          <s:complexType>
            <s:sequence>
              <s:element name="taskID" type="tns:guid" />
              <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetTaskMessageResponse">
          <s:complexType>
            <s:sequence>
              <s:element name="TaskMessage" type="s:string" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="SendTaskResponse">
          <s:complexType>
            <s:sequence>
              <s:element name="taskResponse" type="s:string" />
              <s:element minOccurs="0" maxOccurs="1" name="actingUser" type="s:string" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="SendTaskResponseResponse">
          <s:complexType />
        </s:element>
      </s:schema>
      <s:schema elementFormDefault="qualified"
targetNamespace="http://microsoft.com/wsdl/types/">
        <s:simpleType name="guid">
          <s:restriction base="s:string">
            <s:pattern value="[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-
9a-fA-F]{12}" />
          </s:restriction>
        </s:simpleType>
      </s:schema>
    </wsdl:types>
    <wsdl:message name="GetActivityListSoapIn">
```

```
      <wsdl:part name="parameters" element="tns:GetActivityList" />
    </wsdl:message>
    <wsdl:message name="GetActivityListSoapOut">
      <wsdl:part name="parameters" element="tns:GetActivityListResponse" />
    </wsdl:message>
    <wsdl:message name="AddActionToActivityFlowSoapIn">
      <wsdl:part name="parameters" element="tns:AddActionToActivityFlow" />
    </wsdl:message>
    <wsdl:message name="AddActionToActivityFlowSoapOut">
      <wsdl:part name="parameters" element="tns:AddActionToActivityFlowResponse" />
    </wsdl:message>
    <wsdl:message name="GetActivityFlowInfoSoapIn">
      <wsdl:part name="parameters" element="tns:GetActivityFlowInfo" />
    </wsdl:message>
    <wsdl:message name="GetActivityFlowInfoSoapOut">
      <wsdl:part name="parameters" element="tns:GetActivityFlowInfoResponse" />
    </wsdl:message>
    <wsdl:message name="GetActionInstanceSoapIn">
      <wsdl:part name="parameters" element="tns:GetActionInstance" />
    </wsdl:message>
    <wsdl:message name="GetActionInstanceSoapOut">
      <wsdl:part name="parameters" element="tns:GetActionInstanceResponse" />
    </wsdl:message>
    <wsdl:message name="GetTaskInfoSoapIn">
      <wsdl:part name="parameters" element="tns:GetTaskInfo" />
    </wsdl:message>
    <wsdl:message name="GetTaskInfoSoapOut">
      <wsdl:part name="parameters" element="tns:GetTaskInfoResponse" />
    </wsdl:message>
    <wsdl:message name="GetTaskMessageSoapIn">
      <wsdl:part name="parameters" element="tns:GetTaskMessage" />
    </wsdl:message>
    <wsdl:message name="GetTaskMessageSoapOut">
      <wsdl:part name="parameters" element="tns:GetTaskMessageResponse" />
    </wsdl:message>
    <wsdl:message name="SendTaskResponseSoapIn">
      <wsdl:part name="parameters" element="tns:SendTaskResponse" />
    </wsdl:message>
    <wsdl:message name="SendTaskResponseSoapOut">
      <wsdl:part name="parameters" element="tns:SendTaskResponseResponse" />
    </wsdl:message>
    <wsdl:portType name="HwsServiceSoap">
      <wsdl:operation name="GetActivityList">
        <wsdl:input message="tns:GetActivityListSoapIn" />
        <wsdl:output message="tns:GetActivityListSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="AddActionToActivityFlow">
        <wsdl:input message="tns:AddActionToActivityFlowSoapIn" />
        <wsdl:output message="tns:AddActionToActivityFlowSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetActivityFlowInfo">
        <wsdl:input message="tns:GetActivityFlowInfoSoapIn" />
        <wsdl:output message="tns:GetActivityFlowInfoSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetActionInstance">
        <wsdl:input message="tns:GetActionInstanceSoapIn" />
        <wsdl:output message="tns:GetActionInstanceSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetTaskInfo">
```

```
        <wsdl:input message="tns:GetTaskInfoSoapIn" />
        <wsdl:output message="tns:GetTaskInfoSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetTaskMessage">
        <wsdl:input message="tns:GetTaskMessageSoapIn" />
        <wsdl:output message="tns:GetTaskMessageSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="SendTaskResponse">
        <wsdl:input message="tns:SendTaskResponseSoapIn" />
        <wsdl:output message="tns:SendTaskResponseSoapOut" />
      </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="HwsServiceSoap" type="tns:HwsServiceSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
    <wsdl:operation name="GetActivityList">
      <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityList" style="document"
/>
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddActionToActivityFlow">
      <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/AddActionToActivityFlow"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetActivityFlowInfo">
      <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActivityFlowInfo"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetActionInstance">
      <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetActionInstance"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetTaskInfo">
```

```
        <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskInfo" style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="GetTaskMessage">
        <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/GetTaskMessage" style="document"
/>
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="SendTaskResponse">
        <soap:operation
soapAction="http://microsoft.com/Biztalk2004/Hws/Hwsservice/SendTaskResponse"
style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
   </wsdl:binding>
</wsdl:definitions>
```

7   Appendix B: Message Schemas

7.1   Activation Message Schema

The following is the schema for an Activation Message.

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema
xmlns:ns1="http://schemas.microsoft.com/Hws/2003/HwsPromotedProperties/TemplateLogic"
xmlns:ns0="http://schemas.microsoft.com/Hws/2003/HwsPromotedProperties/ProtocolMessageCorrela
tion" xmlns="http://tempuri.org/Hws_Activate_Sample"
xmlns:b="http://schemas.microsoft.com/BizTalk/2003"
targetNamespace="http://tempuri.org/Hws_Activate_Sample"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <xs:complexType>
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element name="HwsSection">
          <xs:complexType>
            <xs:sequence minOccurs="1" maxOccurs="1">
              <xs:element name="ActivityFlowID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActivityFlowDescription"
type="xs:string">
                <xs:annotation>
                  <xs:appinfo>
                    <b:fieldInfo notes="ActivityFlowDescription is taken from the first
activate message in a new activity flow.  This value is not updated if changed in subsequent
activation messages." xmlns:b="http://schemas.microsoft.com/BizTalk/2003" />
                  </xs:appinfo>
                </xs:annotation>
              </xs:element>
              <xs:element minOccurs="1" maxOccurs="1" name="InitiatingActor" type="xs:string"
/>
              <xs:element minOccurs="1" maxOccurs="1" name="ActionTypeID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActionInstanceID"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActionInstanceDescription"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ParentActionInstanceID"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ParentTaskID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActivityModelTypeID"
type="xs:string" />
              <xs:element name="ActivityModelInstanceID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" default="0" name="ActivityModelStepID"
type="xs:int" />
              <xs:element minOccurs="1" maxOccurs="1" default="false"
name="IsDependentOnParent" type="xs:boolean" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActivityFlowProperties">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element minOccurs="0" maxOccurs="unbounded" name="Property">
                      <xs:complexType>
                        <xs:simpleContent>
                          <xs:extension base="xs:string">
                            <xs:attribute name="Name" type="xs:string" />
                            <xs:attribute name="Description" type="xs:string" />
                            <xs:attribute name="Type" type="xs:string" />
                          </xs:extension>
```

```xml
                        </xs:simpleContent>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              <xs:element minOccurs="1" maxOccurs="1" name="ActionProperties">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element minOccurs="0" maxOccurs="unbounded" name="Property">
                      <xs:complexType>
                        <xs:simpleContent>
                          <xs:extension base="xs:string">
                            <xs:attribute name="Name" type="xs:string" />
                            <xs:attribute name="Description" type="xs:string" />
                            <xs:attribute name="Type" type="xs:string" />
                          </xs:extension>
                        </xs:simpleContent>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              <xs:element minOccurs="1" maxOccurs="1" name="HwsWebServiceUrl"
  type="xs:string" />
            </xs:sequence>
            <xs:attribute fixed="Hws_Activate" name="HwsMessageType" type="xs:string"
  use="required" />
          </xs:complexType>
        </xs:element>
        <xs:element name="ActionSection">
          <xs:complexType />
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="Payloads">
          <xs:complexType>
            <xs:sequence>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="Payload" nillable="true">
                <xs:complexType mixed="true">
                  <xs:sequence>
                    <xs:any minOccurs="0" maxOccurs="unbounded" />
                  </xs:sequence>
                  <xs:attribute name="ID" type="xs:string" />
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  </xs:sequence>
  </xs:complexType>

</xs:schema>
```

7.2 Task Message Schema

The following is the full schema for a Task Message and a Task Response Message.

```xml
<?xml version="1.0" encoding="utf-16"?>
```

```xml
<xs:schema xmlns:hws="http://schemas.microsoft.com/Hws/2003"
xmlns="http://tempuri.org/Hws_Task_Sample"
xmlns:b="http://schemas.microsoft.com/BizTalk/2003"
targetNamespace="http://tempuri.org/Hws_Task_Sample"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:appinfo>
      <hws:schemaInfo description="[default task schema description]" />
      <schemaEditorExtension:schemaInfo namespaceAlias="hws"
extensionClass="Microsoft.BizTalk.Hws.Tools.SchemaEditorExtension" standardName="Hws File"
xmlns:schemaEditorExtension="http://schemas.microsoft.com/BizTalk/2003/SchemaEditorExtensions
" />
      <b:imports xmlns:b="http://schemas.microsoft.com/BizTalk/2003">
        <b:namespace prefix="ns0"
uri="http://schemas.microsoft.com/Hws/2003/HwsPromotedProperties/ProtocolMessageCorrelation"
location="Microsoft.BizTalk.Hws.HwsPromotedProperties.ProtocolMessageCorrelation" />
      </b:imports>
      <b:schemaInfo root_reference="HwsMessage"
xmlns:b="http://schemas.microsoft.com/BizTalk/2003" standard="Hws File" />
    </xs:appinfo>
  </xs:annotation>
  <xs:element name="HwsMessage">
    <xs:annotation>
      <xs:appinfo>
        <b:properties xmlns:b="http://schemas.microsoft.com/BizTalk/2003">
          <b:property name="ns0:TaskID" xpath="/*[local-name()='HwsMessage' and namespace-
uri()='http://tempuri.org/Hws_Task_Sample']/*[local-name()='HwsSection']/*[local-
name()='TaskID']" />
          <b:property name="ns0:ScheduleInstanceID" xpath="/*[local-name()='HwsMessage' and
namespace-uri()='http://tempuri.org/Hws_Task_Sample']/*[local-name()='HwsSection']/*[local-
name()='ActionInstanceID']" />
        </b:properties>
      </xs:appinfo>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence minOccurs="1" maxOccurs="1">
        <xs:element minOccurs="1" maxOccurs="1" name="HwsSection">
          <xs:complexType>
            <xs:sequence minOccurs="1" maxOccurs="1">
              <xs:element minOccurs="1" maxOccurs="1" name="ActivityFlowID" type="xs:string"
/>
              <xs:element minOccurs="1" maxOccurs="1" name="TaskID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="TaskDescription" type="xs:string"
/>
              <xs:element minOccurs="1" maxOccurs="1" name="ActionTypeID" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActionInstanceID"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="InitiatingActor" type="xs:string"
/>
              <xs:element minOccurs="1" maxOccurs="1" name="ActorElementXPath"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="TargetActor" type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" name="ActivityModelTypeID"
type="xs:string" />
              <xs:element minOccurs="1" maxOccurs="1" default="0" name="ActivityModelStepID"
type="xs:int" />
              <xs:element minOccurs="1" maxOccurs="1" name="TaskProperties">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element minOccurs="0" maxOccurs="unbounded" name="Property">
                      <xs:complexType>
```

```xml
                    <xs:simpleContent>
                      <xs:extension base="xs:string">
                        <xs:attribute name="Name" type="xs:string" />
                        <xs:attribute name="Description" type="xs:string" />
                        <xs:attribute name="Type" type="xs:string" />
                      </xs:extension>
                    </xs:simpleContent>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element minOccurs="1" maxOccurs="1" name="ActivityModelInstanceID"
type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="HwsWebServiceUrl"
type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="TaskStatus">
            <xs:simpleType final="restriction">
              <xs:restriction base="xs:string">
                <xs:enumeration value="NotStarted" />
                <xs:enumeration value="InProgress" />
                <xs:enumeration value="Completed" />
                <xs:enumeration value="Waiting" />
                <xs:enumeration value="Deferred" />
                <xs:enumeration value="Accepted" />
                <xs:enumeration value="Tentative" />
                <xs:enumeration value="Declined" />
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
          <xs:element minOccurs="1" maxOccurs="1" default="0" name="PercentageComplete"
type="xs:nonNegativeInteger" />
          <xs:element minOccurs="1" maxOccurs="1" name="NumberOfResponses"
type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Reassignment">
            <xs:complexType>
              <xs:sequence minOccurs="0" maxOccurs="1">
                <xs:element minOccurs="1" maxOccurs="1" name="ReassignedByActor"
type="xs:string" />
                <xs:element name="FromTaskID" type="xs:string" />
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:attribute fixed="Hws_Task" name="HwsMessageType" type="xs:string"
use="required" />
      </xs:complexType>
    </xs:element>
    <xs:element name="ActionSection">
      <xs:complexType />
    </xs:element>
    <xs:element minOccurs="1" maxOccurs="1" name="Payloads">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="Payload" nillable="true">
            <xs:complexType mixed="true">
              <xs:sequence>
                <xs:any minOccurs="0" maxOccurs="unbounded" processContents="skip" />
              </xs:sequence>
```

```
                        <xs:attribute name="ID" type="xs:string" />
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
  </xs:sequence>
 </xs:complexType>
 </xs:element>
</xs:schema>
```

8 Appendix C: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® BizTalk® Server 2004

- Microsoft® Office InfoPath® 2007

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

9 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

10 Index

Release: Sunday, December 19, 2010