

ModusOne

Serviços da Web do ModusOne

Version: 5.2



perceptivesoftware

Written by: Product Documentation, R&D
Date: February 2013

© 2012 Perceptive Software. All rights reserved

ModusOne é uma marca registrada da Lexmark International Technology SA, registrada nos EUA e em outros países.

Perceptive Software é uma unidade de negócios autônoma da Lexmark International Technology SA. Todos os outros nomes de produtos e marcas mencionados neste documento são marcas comerciais ou registradas de seus respectivos proprietários.

Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação ou transmitida por qualquer outra forma ou por qualquer outro meio, eletrônico, mecânico, fotocópia, gravação ou qualquer outra manifestação de mídia hoje conhecida ou que se torne conhecida no futuro, sem a anterior permissão por escrito da Perceptive Software.

Conteúdo

Introdução	1
Arquitetura.....	1
Módulos.....	1
<i>Processo do ModusOne WS</i>	1
Definições.....	1
Identificador do processo	1
Opções de processo	2
Endereço de um objeto ModusOne em um pacote.....	4
Ações, comandos e seus resultados	4
Informações do processo.....	7
Informações sobre encaminhamento de processo	11
Estado da propriedade da ação.....	12
Propriedade da ação configurada.....	12
Atributos especiais da ação PRINTDOCUMENTS.....	13
Atributos especiais da ação DATASELECTION	13
Obter e definir variáveis de pool	14
Exemplos de XML	15
Opções ao fechar um processo	15
Opções ao encaminhar um processo	15
Opções XML de Processlist	16
XML de Processlist.....	19
XML do objeto	20
Ações incorporadas no XML do objeto.....	24
Informações sobre XML	28
Logon XML	29
XML de UserInfo.....	29
Erro XML.....	32
Funções.....	32
Limpar cache.....	32
Doc_GetFile_Mime	33
Doc_SetFile_Mime.....	34
Logon.....	34
Logoff.....	35

Obj_GetStructure.....	35
Obj_SetStructure.....	36
Obj_Toggle.....	37
Ping.....	37
Process_Close.....	38
Process_Create.....	39
Process_Delete.....	40
Process_Forward.....	41
Process_GetInfo.....	42
Process_GetLastError.....	42
Process_GetList.....	43
Process_Open.....	44
Process_SetData.....	44
Process_Start.....	45
<i>Repositório do ModusOne WS.....</i>	<i>46</i>
Exemplos de XML.....	46
XML de Systemdef.....	46
MwsItemInfo XML.....	46
XML MwsSystemList.....	47
XML MwsNavigTree.....	47
MwsNavigList XML.....	47
MwsFolderContent XML.....	49
MwsLookupResult XML.....	49
MwsDataProviderDefinition XML.....	49
XML MwsValueHelpDefinition.....	50
Opções XML de MwsNavigTree.....	50
Funções.....	50
Ping.....	50
Rep_GetBinFile_Mime.....	51
Rep_GetDataProviderDefinition.....	51
Rep_GetFolderContent.....	52
Rep_GetForms.....	53
Rep_GetItemInfo.....	53
Rep_GetLookupObjects.....	54

Rep_GetLookupValue.....	55
Rep_GetNavigList.....	56
Rep_GetNavigTree	56
Rep_GetPrinters	57
Rep_GetSystem.....	58
Rep_GetSystemList.....	58
Rep_GetValueHelpDefinition	58
<i>ModusOne WS UserRepository</i>	<i>59</i>
Exemplos de XML	59
RoleList XML	59
XML de UserList para GetUsers.....	59
XML de UserList para GetUsersOfRole	60
Funções.....	60
GetRoles	60
GetUsersOfRole.....	60
GetUsers.....	61
Ping.....	61
<i>Valores de retorno</i>	<i>61</i>
Valores de retorno de funções do MWS.....	61
Resultado da execução com êxito	62
Códigos de erro	62
Avisos.....	62
Informações adicionais	62
<i>Integração.....</i>	<i>62</i>
<i>Repositório do ModusOne</i>	<i>63</i>
<i>Processos de correspondência.....</i>	<i>63</i>

Introdução

Os Serviços da Web do ModusOne Web (MWS) requerem uma instalação do ModusOne totalmente funcional.

Eles fornecem aplicativos de terceiros com várias funções de um sistema de correspondência por meio de mensagens SOAP (Simple Object Access Protocol).

Os serviços da Web do ModusOne permitem aos aplicativos de terceiros:

- Abrir e fechar sessões no lado do servidor
- Recuperar informações de um repositório do ModusOne
Para obter mais informações, consulte o capítulo sobre o Repositório do ModusOne.
- Iniciar e gerenciar processos de correspondência
Para obter informações mais detalhadas, consulte o capítulo sobre Processos de correspondência.

Os aplicativos da Web convencionais e clientes finos (ou seja, aplicativos Windows que acessam a interface SOAP diretamente) podem ser desenvolvidos com base nos serviços da Web.

Para obter informações mais detalhadas, consulte o capítulo sobre [Integração](#).

Arquitetura

De um ponto de vista técnico, os serviços da Web do ModusOne podem ser divididos em três camadas:

- A camada da interface é responsável por publicar os serviços da Web do ModusOne e controlar o acesso a eles. A comunicação com o mundo externo é feita exclusivamente por mensagens SOAP.
- A lógica de negócios de processos correspondentes está situada na camada de negócios. É neste local que ocorrem o acesso ao repositório do ModusOne e a criação do documento real. A comunicação com o servidor de dados do ModusOne também é responsabilidade da camada de negócios. Assim como no ModusOne Studio, é possível executar seleções de dados definidos em Xdata ou importar arquivos XML.
- A camada de estado é responsável por salvar todos os dados relacionados ao repositório, ao status e ao processo, fazendo dela a base uniforme para administração de texto e funcionalidade do usuário corporativo.

Módulos

Processo do ModusOne WS

Definições

Identificador do processo

O PID (Identificador do processo) é um número único do processo.

Todos os dados do processo estão armazenados sob este número no servidor. Sendo assim, ele deve ser um número único para todo o sistema.

O PID pode ter no máximo 40 caracteres e deve utilizado como um nome de arquivo.

Observe:

Um processo deve sempre ser excluído com a função **Process_Delete** para liberar os recursos no servidor.

Depois de chamar **Process_Delete**, o **PID** não poderá mais ser usado.

Opções de processo

As opções são usadas para definir várias configurações necessárias para um processo individual. Elas devem sempre ser passadas como string de XML ampla.

A estrutura da string de XML se parece com o resultado XML da função **Process_GetInfo** e reflete a estrutura interna do objeto do MWS.

Consulte também:

Informações sobre XML

Observe:

Ao configurar opções, somente atributos explicitamente declarados como atributos nesta seção podem ser definidos.

Como uma string de XML ampla é usada, é possível incluir várias configurações diferentes de uma vez.

Essencialmente, são elas:

- Definir o comando de início
Consulte também:
 Função **Process_Create**, elemento **mwsprocessmngr**, atributo **onstart**.
 Valores de atributo possíveis: ações, comandos e seus resultados
- Definir título e descrição do processo
Consulte também:
 Função **Process_Close** ou Opções ao fechar um processo.
- Definir configurações gerais de ações
 - Cada ação de um processo pode ser configurada.
 - Há ações que requerem uma configuração e ações que não necessariamente devem ser configuradas.
 - Para todas as ações (elemento **mwsaction**), os atributos **state** e **configured** podem ser definidos.
 - A configuração da propriedade da ação **state** de **finished** para o valor **ready** significa que ação pode ser repetida.
 Observe que ao fazer isso, qualquer resultado existente dessa ação (por exemplo, documentos) e os resultados de todas as ações subsequentes são excluídos.
 - A configuração da propriedade da ação **configured** como verdadeiro estipula que a configuração dessa ação está concluída.
- Define parâmetros para recuperação de dados
 - Os parâmetros para recuperação de dados pertencem à ação **DATASELECTION** e devem, portanto, ser definidos como elemento filho dessa ação.
 - O nome do elemento definido para todos os parâmetro de recuperação de dados é **selparams**.
 - Os parâmetros são endereçados pelo atributo **name**. O valor deve ser passado como CDATA.

Exemplo:

```

...
<selparams>
  <selparam name="Partner-Nr">0815</selparam>
</selparams>
...

```

- Definir variáveis manuais
 - As variáveis manuais pertencem à ação **CREATEDOCUMENTS** e devem, portanto, ser definidas como elemento filho dessa ação.
 - O nome do elemento definido para todas as variáveis manuais de um pacote é **manvars**.
 - O nome do elemento definido para todas as variáveis manuais de um documento é **docref**.
 - O documento deve ser endereçado pelo atributo **reference** (consulte: Endereço de um objeto ModusOne em um pacote).
 - A variável manual real é descrita pelo nome **manvar** no elemento e também deve ser endereçada pelo atributo **reference**. O valor deve ser passado como CDATA.

Exemplo:

```

...
<manvars>
  <docref reference="790B817A-B2C3-475E-86E5-08118150EA94">
    <manvar reference="DocumentCollection.790B817A-B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_E4437CCD-E71A-4181-9C6E-0691B26E0C8F">Value 1</manvar>
    <manvar reference="DocumentCollection.790B817A-B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_EAAC39D9-81BA-476B-A003-039ABF3618AA" >Value 2</manvar>
  </docref>
</manvars>
...

```

- Definir parâmetros para gerenciamento de saída

Os parâmetros para gerenciamento de saída pertencem à ação **PRINTANDARCHIVE** e devem, portanto, ser definidos como atributos dessa ação.

Consulte também:

Atributos especiais da ação PRINTDOCUMENTS

- Definir variáveis no pool SYSTEMPOOL

As variáveis de pool são independentes de ações e podem, portanto, ser definidas diretamente abaixo do elemento 'process'.

Consulte também:

Obter e definir variáveis de pool

Como regra geral, os atributos a seguir não podem ser definidos:

- process_status
- lastaction
- lastactionname

- o lasterrorcode
- o lasterrormessage

Endereço de um objeto ModusOne em um pacote

Cada objeto ModusOne em um pacote tem uma referência exclusiva.

Essa referência é definida no atributo **reference**. Uma referência é composta dos IDs de filhos dos objetos principais e do objeto em si.

Exemplo:

790B817A-B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_EAAC39D9-81BA-476B-A003-039ABF3618AA

Ações, comandos e seus resultados

Um processo de serviço da Web do ModusOne Web é definido essencialmente por suas ações.

Uma ação pode ter um antecessor e pode requerer configuração.

Além disso, uma ação tem um status que fornece informações sobre o progresso da ação.

Os componentes do processo MwsActivityStatus em um processo MWS servem para representar ações.

Ações do serviço da Web do ModusOne Web ao usar o processo padrão MWS

OBJECTSELECTION	Seleção de objetos
DATASELECTION	Recuperação de dados
COMPLETESTRUCTURE	Estrutura em pacote completa, por exemplo, pacote dinâmico
CREATEDOCUMENTS	Geração de documentos
FORWARD	Encaminhar processo
PRINTDOCUMENTS	Imprimir documentos em pacote
PRINTANDARCHIVE	Passar documentos em pacote para Odin

Esta sequência também define as dependências entre ações individuais:

- Só é possível imprimir depois que os documentos foram criados.
- Os documentos só podem ser criados quando está claro quais documentos deverão ser criados.
- Só fica claro quais documentos devem ser criados depois que os dados forem recuperados.

Além dessas dependências fixas entre ações, há outras dependências dinâmicas que nem sempre estão presentes e, com certa frequência, só podem ser resolvidas por interação do usuário.

Em outras palavras, em alguns casos, as ações só poderão ser executadas quando a ação tiver sido configurada.

Dependências dinâmicas

- Os dados só podem ser recuperados quando parâmetros de seleção tiverem sido definidos. A necessidade de inserir parâmetros, entretanto, só é determinada quando esses parâmetros realmente existem.
- Os documentos só podem ser criados quando suas variáveis manuais tiverem sido definidas. A necessidade de inserir variáveis manuais, entretanto, só é determinada quando variáveis manuais realmente existem. Isso pode, obviamente, mudar muito rapidamente, dependendo de quais blocos de texto foram selecionados.
- Um pacote só pode ser impresso diretamente quando uma impressora direta tiver sido configurada. A necessidade de inserir uma impressora direta, entretanto, só é determinada quando a impressão direta é exigida.
- Um pacote só pode ser encaminhado quando o usuário e/ou um grupo de usuários aonde ele deverá ser encaminhado tiver sido definido. Se uma ou mais dessas dependências dinâmicas estiverem presentes, a respectiva ação só será iniciada com êxito quando a configuração estiver concluída. Uma ação é considerada para ser configurada quando a propriedade (consulte Propriedade da ação configurada) foi definida corretamente. As propriedades da ação podem ser passadas nas opções (consulte Opções de processo) ao chamar as funções **Process_Start** ou **Process_Create**.

Controle do processo

O controle do processo de serviços da Web do ModusOne estipula que, basicamente, toda ação sempre pode ser iniciada.

Se os antecessores da ação a ser iniciada ainda não foram executados, o controle do processo tenta iniciá-los. A verificação das ações anteriores sempre se inicia como a primeira ação do processo.

Se, por exemplo, logo após a criação, um processo **PRINTDOCUMENTS** for iniciado, o controle do processo tentará executar todas as ações pendentes iniciando com a primeira ação do processo.

Se uma dessas ações não estiver configurada, um aviso da configuração correspondente será emitido. Caso contrário, a ação é executada e a próxima ação da sequência é verificada.

O processo continua até o primeiro aviso de configuração ou termina com a execução da ação de destino iniciada, neste caso, **PRINTDOCUMENTS**.

Manipulação de erro

Se for necessária entrada manual, iniciar um comando resultará no retorno do código de erro 6000.

O MWS armazena o comando originalmente iniciado para que possa ser continuado depois que as informações ausentes forem inseridas. No próximo início do comando, o comando armazenado pode ser reativado pelo comando fictício **CONTINUE_LAST_ACTION**.

Exemplo de pseudocódigo:

```
...
res := Process.Process_Start(FSID, FPID, 'PRINT_AND_ARCHIVE', options, resxml
);
```

```

if res = 6000 then
begin
    ok := LetUserConfigActions(options, resxml);
    if ok then
        res := Process.Process_Start(FSID, FPID, 'CONTINUE_LAST_ACTION', options, resxml);
end;
...

```

Comandos e seus resultados

Como visto no último exemplo, o conteúdo do parâmetro **command** não está consistente com o nome das ações do processo.

Com exceção dos dois comandos estáticos **SET_PARAMETERS** e **CONTINUE_LAST_ACTION**, os comandos são derivados dos nomes dos blocos do processo de execução de script contidos no processo MWS.

Aqui, a atividade de destino é definida ao especificar o comando do primeiro bloco do processo de execução de script cujo nome do objeto corresponde ao comando passado.

Se nenhum bloco do processo de execução de script for encontrado, um erro correspondente será lançado para esse efeito.

A lista a seguir aplica-se ao processo padrão MWS.

- **SET_PARAMETERS**
Esse comando avalia e define os parâmetros nas opções. Uma ação não foi iniciada. Uma string vazia será retornada se tiver êxito.
- **SELECT_OBJECT**
A ação OBJECTSELECTION é iniciada. Se tiver êxito, o resultado XML será semelhante ao do comando **SELECT_DATA**.
Se nenhum pacote estiver carregado e os parâmetros correspondentes para carregar o objeto (pacote) forem exigidos, o comando retornará o código de erro 6000.
Uma descrição da estrutura do XML pode ser encontrada no capítulo sobre Exemplo de XML de um Aviso de configuração para a Ação **FORWARD**.
- **SELECT_DATA**
A ação DATASELECTION (recuperação de dados) é iniciada. Se tiver êxito, um XML será retornado representando a estrutura do pacote. Esse XML corresponderá ao resultado do procedimento **Obj_GetStructure** se uma string vazia for passada no parâmetro **RootRef** e o valor 1 for passado como parâmetro **MaxLevel**.
O comando retorna o código de erro 6000 quando parâmetros são exigidos para iniciar a seleção. Neste caso, o resultado XML contém os parâmetros necessários.
Uma descrição da estrutura do XML pode ser encontrada no capítulo sobre Exemplo de XML para SELPARAMS.
- **COMPLETE_STRUCTURE**
O evento **AfterDataRetrieval** é acionado e o respectivo script (se atribuído) é executado. O resultado XML corresponde ao do comando **SELECT_DATA**.
- **CREATE_DOCUMENTS**
A criação do documento é iniciada. Aqui, como para o comando **SELECT_DATA**, a estrutura do pacote é retornada quando tiver êxito.
O comando retorna o código de erro 6000 quando variáveis manuais ainda forem necessárias.

Neste caso, o resultado XML contém as variáveis manuais necessárias.

Uma descrição da estrutura do XML pode ser encontrada no capítulo sobre Exemplo de XML para MANVARS.

- **FORWARD_DOCUMENTS**

A ação **FORWARD** é executada. Se tiver êxito, o Resultado XML será semelhante ao do comando **SELECT_DATA**. O comando retorna o código de erro 6000 quando o pacote está bloqueado (o sinalizador BlockBundle no script 'AfterCreatingDocuments' atribuído é definido como verdadeiro e o usuário atual é o mesmo usuário que criou o processo). Neste caso, o resultado XML contém os parâmetros necessários para encaminhamento. Uma descrição da estrutura do XML pode ser encontrada no capítulo sobre Exemplo de XML para SELPARAMS.

- **PRINT_DOCUMENTS**

Executa a ação **PRINTDOCUMENTS**.

Prepara a transferência para gerenciamento de saída. Se tiver êxito, o resultado XML será semelhante ao do comando **SELECT_DATA**.

O comando retornará o código de erro 6000 se nem todos os documentos tiverem uma definição de impressão válida. Neste caso, o resultado XML contém os documentos cujas definições de impressão ainda deverão ser concluídas pelo método **Obj_SetStructure**.

Mais informações detalhadas sobre a estrutura do XML podem ser encontradas no capítulo sobre OUTPUTPARAMS.

- **PRINT_AND_ARCHIVE**

A transferência para gerenciamento de saída é realizada pelo componente odinProcessImport. Se tiver êxito, o resultado XML será semelhante ao do comando **SELECT_DATA**. Caso contrário, um erro correspondente de XML será retornado.

- **CONTINUE_LAST_ACTION**

Se um comando foi interrompido pelo código de erro 6000, ele poderá ser reativado, ou seja, reiniciado usando esse comando fictício.

Se nenhum comando armazenado for encontrado, um erro correspondente de XML será retornado.

Observação:

Para garantir que todos os comando sejam executados sem interrupção, os parâmetros de configuração necessários podem, ou respectivamente, devem ser passados nas opções e as ações correspondentes definidas como **configuradas**.

Para obter mais informações, consulte o capítulo sobre [Opções de processo](#).

Informações do processo

Um processo de serviço do ModusOne Web é composto de várias ações, um gerenciador de processos que controla os processos e pools de variáveis. Cada um destes objetos tem propriedades que podem mudar durante o curso de um processo.

Um aplicativo que integra o MWS pode acessar informações sobre um processo para exibi-lo ou, respectivamente, decidir quais ações subsequentes devem ser executadas.

Para manter o tráfego de dados de rede a um mínimo, é possível solicitar essa informação seletivamente. Para fazer isso, o tipo de informação deve ser especificado.

As propriedades do gerenciador do processo são sempre retornadas, independentemente do tipo de informação solicitada. O elemento mwsprocessmngtr está localizado abaixo do processo do elemento.

Exemplo

```
...
  <mwsprocessmngr lasterrorcode="0" lasterrormessage="" lastactionname="" on
start="">
```

Atributo	Descrição
lasterrorcode	Número do último erro
lasterrormessage	Descrição do último erro
onstart	Nome da ação a ser iniciada diretamente após a criação de um processo. Observação A configuração desta propriedade em um ponto após a criação do processo não tem efeito!
lastactionname	Nome da última ação executada

Outra informação importante é o último erro que ocorreu. Essa informação é sempre retornada quando um erro foi lançado, independentemente do tipo de informação solicitada. A informação está localizada logo abaixo do elemento mws.

Exemplo

```
...
  <error errorcode="6000">error text</error>
```

Os tipos de informação a seguir são suportados no momento:

PROCESSINFO

Atributo	Descrição
Cargo	Título do processo
Descrição	Descrição do processo

As ações são listadas logo abaixo do gerenciador do processo no elemento **mwsaction**.

Cada ação tem os seguintes atributos:

Atributo	Descrição
name	Nome da ação
lasterrorcode	Número do último erro

lasterrormessage	Descrição do último erro
state	Consulte o estado da Propriedade da ação
configured	Consulte a Propriedade da ação configurada

Atributos estendidos da ação **DATASELECTION**:

Consulte Atributos especiais da ação DATASELECTION:

Atributos estendidos da ação **PRINTANDARCHIVE**:

Consulte Atributos especiais da ação PRINTDOCUMENTS

POOLVARS

Retorna variáveis **SYSTEMPOOL**: Consulte: Obter e definir variáveis de pool.

MANVARS

Retorna variáveis manuais de um pacote.

Como as variáveis manuais pertencem à ação **CREATEDOCUMENTS**, elas são listadas abaixo do elemento correspondente.

Exemplo

```

...
<mwsaction name="CREATEDOCUMENTS" lasterrorcode="0" lasterrormessage="" state
="0" configured="1">
  <manvars>
    <docref name="Partner_Application_ManVar" title="Partner_Application_Ma
nVar" reference="790B817A-B2C3-475E-86E5-08118150EA94">
      <manvar name="Agent_Forename" reference="DocumentCollection.790B817A
-B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_E4437CCD-
E71A-4181-9C6E-
0691B26E0C8F" title="Agent_Forename" mask="!90/90/0000;1;_" vh_name="" vh_id=
"" vh_system="" input_enabled="Y" data_link="MODUSUSER" data_attribute="Foren
ame" input_required="0" >Value1</manvar>
      <manvar name="Agent_Surname" reference="DocumentCollection.790B817A-
B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_EAAC39D9-
81BA-476B-A003-
039ABF3618AA" title="Agent_Surname" mask="90/90/0000;1;_" vh_id="" vh_system
="" input_enabled="Y" data_link="MODUSUSER" data_attribute="Surname" input_re
quired="0" >Value2</manvar>
    </docref>
  </mwsaction>
...

```

O elemento **manvars** é um nó coletor para todas as variáveis manuais. As variáveis manuais de um documento são listadas sob o nó coletor docref.

Este elemento tem os seguinte atributos:

Atributo	Descrição
----------	-----------

name	Nome do documento
title	Título do documento
reference	Consulte Endereço de um objeto ModusOne em um pacote

O elemento manvar descreve uma variável manual conforme a seguir:

Atributo	Descrição
name	Nome da variável manual
reference	Consulte Endereço de um objeto ModusOne em um pacote
title	Título do prompt
mask	Máscara de entrada (consulte a documentação do ModusOne)
vh_name	Nome da ajuda de valor (opcional)
vh_id	ID do objeto da ajuda de valor (opcional)
vh_system	ID do sistema da ajuda de valor (opcional)
input_enabled	0 quando nenhuma entrada manual for possível (por exemplo, se apenas a seleção de valor for possível) Este atributo só existirá se nenhuma entrada manual for possível.
Dados	Valor atual da variável manual

SELPARAMS

Retorna parâmetros para recuperação de dados. Como esses parâmetros pertencem à ação DATASELECTION, eles são listados abaixo do elemento correspondente.

Exemplo

```

...
<mwsaction name="DATASELECTION" lasterrorcode="0" lasterrormessage="" state="
0" configured="0">
  <selparams>
    <selections>
      <selection name="Partner_Letter" title="Partner_Letter" active="True
">
        <selparam name="Partnernumber" displayname="PartnerNumberTest" ma
sk="" />0815</selparam>

```

```

        </selection>
    </selections>
</selparams>
</mwsaction>
...

```

Um parâmetro de recuperação de dados é descrito pelo elemento selparam:

Atributo	Descrição
name	Nome do parâmetro
displayname	Nome usado para exibição. Se estiver vazio, o nome será usado.
mask	Máscara de entrada desse parâmetro. Se estiver vazio, nenhuma máscara de entrada será usada.
data	Valor do parâmetro

OUTPARAMS

Retorna parâmetros para gerenciamento de saída. Como esses parâmetros pertencem à ação **PRINTDOCUMENTS**, eles são listados abaixo do elemento correspondente.

Exemplo

```

...
<mwsaction name="PRINTDOCUMENTS" lasterrorcode="0" lasterrormessage="" state="0" configured="0">
    <objects>
        <object title="ApplicationPack" type="D" name="ApplicationPack" index="07102009-289-tdr-dm1" reference="933110B7-0485-4C62-A50C-0B9D7FC62DB1" />
    </objects>
</mwsaction>
...

```

Todos os objetos são listados onde as definições de impressão ainda serão concluídas pelo método `Obj_SetStructure`.

Informações sobre encaminhamento de processo

Este XML é retornado da função **Process_Start** quando o comando **PrintAndArchive** é executado em um pacote bloqueado.

Um pacote é bloqueado quando o sinalizador **BlockBundle** em um script **AfterCreatingDocuments** atribuído é definido como verdadeiro e o usuário atual é o mesmo usuário que criou o processo.

Para ativar um Cliente para realizar encaminhamento automático, um destino pode ser definido no script **AfterCreatingDocuments** (variáveis `DestUser` e/ou `DestUsergroup`).

Os destinos definidos aqui são retornados dos atributos correspondentes para que possam ser usados como parâmetros para a chamada da função **Process_Forward**. Pelo menos um destino é necessário.

Se nenhum destino for definido no script **AfterCreatingDocuments**, o Cliente deverá implementar o respectivo processo de seleção de destino por conta própria.

Exemplo

```
<mws type="FORWARD" version="2">
  <error errorcode="6000">Configuration for action FORWARD required!</error>
  <process id="120de98f-9ec4-469a-b7ad-376651986aa9" systemoid="dm1">
    <title>BasicBindingTest</title>
    <mwsprocessmngr lasterrorcode="6000" lasterrormessage="Configuration fo
r action FORWARD required!" lastactionname="PRINT_AND_ARCHIVE" onstart="">
      <mwsaction name="FORWARD" lasterrorcode="0" lasterrormessage="" stat
e="0" configured="0"><forwardparams destuser="MyDestUser" destusergroup="MyDe
stGroup"/></mwsaction>
    </mwsprocessmngr>
  </process>
</mws>
```

Atributo	Descrição
destuser	O usuário especificado no script AfterCreatingDocuments para o qual o processo deverá ser encaminhado.
destusergroup	O grupo de usuários especificado no script AfterCreatingDocuments para o qual o processo deverá ser encaminhado.

Estado da propriedade da ação

Valores possíveis

Atributo	Descrição
0	desconhecido (nenhuma informação de status disponível)
1	pronto (pronto para executar)
2	ocupado (em execução)
3	concluído (execução concluída)

Propriedade da ação configurada

Valores possíveis

Atributo	Descrição
----------	-----------

0	falso (não configurado)
1	verdadeiro (configurado)

Atributos especiais da ação PRINTDOCUMENTS

A utilização do método Obj_SetStructure permite definir ou alterar a definição de impressão de um documento.

Atributo	<value>	Descrição
print_asktime	0	Consultar tempo de impressão.
	1	Não consultar tempo de impressão.
form		Nome do formulário
printer		Nome da impressora
printer_type	OEP	Impressora de email
	OP	Impressora online
Impressora de email	0	desconhecido
	1	diretamente
	2	direcionado
print_changeable	0	Muda para definição de impressão permitida
	1	Muda para definição de impressão não permitida

Exemplo

```
<object reference="790B817A-B2C3-475E-86E5-08118150EA94" printer="OnlinePrinter" printer_type="OP" form="Form1" copycount="0" copytext="" printtime="2" />
```

Atributos especiais da ação DATASELECTION

Atributo	Descrição
selection	Nome da seleção

Exemplo

```
<selections>
```

```
<selection name="Partner_Letter" title="Partner_Letter" active="True">
  <selparam name="Partnernumber" displayname="PartnerNumberTest" mask=""/>0815</selparam>
</selection>
<selection name="Partner_V2" title="Partner_V2" active="False"/>
</selections>
```

Seleções do elemento	Elemento de coleta de todas as referências/seleções de dados existentes em um pacote		
Seleção do elemento	Descrição de uma referência de dados com os atributos:		
	name	Nome da referência de dados	
	title	Título da referência de dados	
	ativo	verdadeiro	Seleção ativa
		falso	Seleção XML (dados passados ao MWS pelo método SetData)
Elemento selparams	Descrição do parâmetro para seleção ativa com os atributos:		
	name	Nome do parâmetro	
	displayname	Nome de exibição do parâmetro	
	mask	Máscara de entrada	
	DATA	Valor atual do parâmetro, se já estiver definido	

Obter e definir variáveis de pool

As variáveis de pool podem ser consultadas no contexto de recuperação de informações usando o método **Process_GetInfo** (infoType POOLVARS).

Os pools **SYSTEMPOOL** e **MODUSUSER** são retornados.

Somente variáveis do pool **SYSTEMPOOL** podem ser definidas.

As variáveis desses pools são definidas com o método **Process_Start**.

Se somente as variáveis de pool forem definidas e nenhuma outra ação for tomada, o comando **SET_PARAMETERS** deve ser usado.

Exemplo de XML

```
<pools>
  <pool name="SYSTEMPOOL">
```

```

    <poolvar name="MyVariable" type="S">MyVariableValue</poolvar>
  </pool>
</pools>
...

```

Pools do elemento	Nó da coleta para todos os pools		
Pool do elemento	Nó da coleta para todas as variáveis de pool de um pool específico. O nome do atributo contém o nome do pool.		
Poolvar do elemento	Descrição de uma variável de pool com os atributos:		
	name	Nome da variável de pool	
	type	Tipo da variável de pool	B: Booleano (VERDADEIRO ou FALSO) L: Longinteger (Número) S: Widestring (qualquer string) F: Flutuante (número do ponto flutuante, separador decimal de acordo com configurações regionais)
	data	Conteúdo da variável de pool	

Observação

Ao configurar variáveis, o atributo **type** é opcional.

Se o atributo não for definido, o tipo **Widestring** será assumido.

Se um tipo estiver definido, o conteúdo especificado na área de dados deverá corresponder ao tipo de dados.

Exemplos de XML

Opções ao fechar um processo

```

<mws version="2">
  <title>Title of the process</title>
  <description>Description of the process</description>
</mws>

```

Opções ao encaminhar um processo

```

<mws version="2">
  <title>Title of the process</title>
  <description>Description of the process</description>
</mws>

```

Opções XML de Processlist

Estas opções XML podem ser usadas para especificar quais processos incluir na lista de processos recuperados.

As configurações possíveis são:

1. Retorna os processos criados pelo usuário (padrão quando string vazia é passada).
2. Retorna os processos encaminhados ao usuário diretamente ou a um grupo ao qual o usuário pertence.
3. Retorna processos próprios e processos encaminhados. Se uma string vazia for passada para este parâmetro, somente processos do próprio usuário serão retornados (consulte 1. acima).

```
<mws>
  <processlist>
    <item name="forwarded" include="1" />
    <item name="nonforwarded" include="0"/>
  </processlist>
</mws>
```

Os itens do elemento 'processlist' especificam quais processos são incluídos na lista de processos retornados.

Os atributos têm o seguinte significado:

Atributo	<value>	Descrição
name	forwarded	Retorna todos os processos encaminhados. Esses são os processos encaminhados ao usuário diretamente ou a um grupo ao qual o usuário pertence.
include	0 1	Os processos correspondentes não são incluídos na lista de processos retornados. Os processos especificados pelo atributo name estão incluídos na lista.
name	nonforwarded	Retorna todos os processo que não foram encaminhados. Esses são os processos que o usuário criou.
include	0 1	Os respectivos processos não são incluídos na lista de processos retornados. Os processos especificados pelo atributo 'name' estão incluídos na lista.

Recursos especiais

Se uma string vazia for passada para as opções ou ambos os atributos estiverem desabilitados (include="0"), a configuração padrão de processos únicos criados pelo usuário será retornada.

Atributos XML opcionais estendidos

Os atributos a seguir podem ser definidos opcionalmente para especificar ainda mais a lista de processos retornados:

```
<mws>
  <processlist>
    ...
    <item name="listuser" include="Admin" />
    <item name="shortprocdesc" include="10"/>
    <item name="additionalCondition" include=" AND (MWS_STATUS = 2)" />
    <item name="replacecondition" include="(MWS_STATUS = 2)" />
    ...
  </processlist>
</mws>
```

Os atributos têm o seguinte significado:

Atributo	<value>	Descrição
name	listuser	Se este atributo estiver definido, uma lista do respectivo processo para o usuário especificado será retornada. Na lista de processos encaminhados, somente processos que foram encaminhados ao usuário especificado diretamente são encaminhados.
include	Nome de usuário	Este atributo será ignorado se contiver uma string vazia.
name	shortprocdesc	Se este atributo for especificado e um processo tiver uma descrição do processo, o elemento do processo dentro do XML de ProcessList retornado conterá o atributo shortprocdesc . Este atributo contém os primeiros X caracteres da descrição do processo. Se nenhuma descrição do processo for encontrada, o atributo correspondente não será incluído. Exemplo <process id="ade9e10d-ccfb-490f-a5b4-6fb1ac69bfd8" user="Admin" system="dml" type="" objectname="TestBundle" title="My Title" state="0" created_at="27012010140555" lastsaved="04022010173422" created_by="Admin" shortprocdesc="ShortDescription" />
include	Número de caracteres	Número de caracteres a ser retornado da descrição do processo. Este atributo será ignorado se contiver o valor 0.
name	additionalcondition	Se este atributo for definido, a condição da instrução selecionado

		será expandida para incluir a condição especificada. Desse modo, as condições que se aplicam como resultado de outras opções da lista de processos podem ser expandidas individualmente.
include	A condição a ser adicionada	Este atributo será ignorado se contiver uma string vazia.
name	replacecondition	Se este atributo for definido, a condição da instrução selecionada será completamente substituída pela condição especificada aqui. Todas as outras opções da lista de processos são substituídas. Desse modo, é possível retornar exatamente aqueles processo na lista de processos que correspondem à condição especificada.
include	Nova condição	Este atributo será ignorado se contiver uma string vazia.

Exemplo de uso:

Neste exemplo, todos os processos que estão concluídos (status = 2) ou bloqueados (status = 1) e não foram editados por pelo menos um dia são excluídos.

Para fazer isso, a opção da lista de processos 'replacecondition' é usada para redefinir completamente a condição da instrução selecionada e, subseqüentemente, excluir todos os processos retornados na lista de processos.

```

;***** MLMwsClient *****
GetObject("MwsClient", "MLMwsClient")
MwsClient.MwsUrl = "http://localhost:8011/mws/mwsprocess"
GetObject("date", "MLDate")
;
; Current date
CurrDateAsInt = date.CurrentDate()
;
; current date minus 1 day
SearchDateInt = date.IncDate(CurrDateAsInt, -1, 0, 0)
;
;SearchDate as string
SearchDateStr = date.Int64ToDateString("dd.MM.yyyy", SearchDateInt)
FreeObject("date")
Protocol(SearchDateStr, 8)
;
; replacecondition --> Status = 1 and last edit less than SearchDate
options = FormatStr("<mws><processlist><item name=\"replacecondition\" include=\"(MWS_STATUS = 2) OR ((MWS_STATUS = 1) AND (MWS_LASTSAVED <= '%s'))\" /></processlist></mws>", SearchDateStr)
myXmlResult = ""
res = MwsClient.Process_GetList(0, 100, options)
;

```

```

Protocol(res, 8)
myXmlResult = MwsClient.LastXmlResult
Protocol(result, 8)
;
if (myXmlResult <> "")
  GetObject("xml", "MLXmlDocument")
  xml.LoadXml(myXmlResult)
  root = xml.DocumentElement ;rootElement holen
  nodeList = root.SelectNodes("/mws/processlist/process") ;get all nodes of
the process list
  Protocol("Number of processes: {0}", 8, nodeList.Count)
  nodeListEnumerator = nodeList.GetEnumerator() ; Get an enumerator for the
nodes
  ;
  ; Iterate over all items...
  readNext = nodeListEnumerator.MoveNext()
  while (readNext)
    node = nodeListEnumerator.Current ; get current item node
    ;
    ;read processid
    attrColl = node.Attributes
    attr = attrColl.GetNamedItem("id")
    processId = attr.Value
    Protocol("ProcessId: {0}", 8, processId)
    if (processId <> "")
      MwsClient.Process_Delete(processId) ; Delete process
      Protocol("Process [{0}] deleted", 8, processId)
    end-if
    readNext = nodeListEnumerator.MoveNext() ; get next item node
  end-while
  FreeObject("xml")
end-if
FreeObject("MwsClient")
;

```

XML de Processlist

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="PROCESSLIST" ver
sion="2.0" eof="true" from="0" to="100">
  <processlist>
    <process id="5002f2fb-0d6d-4c30-96f4-
2015785854e8" user="MMueller" system="dml" type="mmtype" objectname="Applicat
ionPack_Perf" state="1" created_at="13102009114113" lastsaved="13102009114119
" created_by="MMueller" />
    <process id="d381cb40-aa53-4443-9683-
542b8cc557cf" user="MMueller" system="dml" objectname="ApplicationPack" title

```



```
= "Adminstrator_Role" state="0" created_at="13102009114006" lastsaved="13102009114015" created_by="MMueller" />
</processlist>
</mws>
```

Descrição do elemento 'process':

Atributo	Descrição
id	ID do processo
user	Nome de usuário
system	OID de sistema do sistema de onde o pacote foi carregado
type	Tipo do processo especificado (contido somente se type estiver definido)
objectname	Nome do objeto carregado
title	Título do processo especificado (contido somente se title estiver definido)
state	Status do processo
created_at	Data em que o processo foi criado. Formato de data: DDMMAAAHHMMNN
lastsaved	Data da última atividade do usuário para este processo. Formato de data DDMMAAAHHMMNN
created_by	Nome do usuário que criou o processo.

XML do objeto

```
<mws type="OBJECT">
  <process id="52a8ce36-0885-4bed-89e6-9ec6357a6af5">
    <object title="ApplicationPack" type="P" name="ApplicationPack" index="02062008-822-tdc-itl1" reference="">
      <object title="Partner_ApplicationPack" type="D" name="Partner_ApplicationPack" in dex="02062008-823-tdr-itl1" reference="10DE6C09-79CF-4547-97AA-6302CF976185" state="1" enabled="1" print_asktime="0" form="Formular1" printer="" printer_t ype="" printtime="2" print_changeable="1" copycount="0" copy="0" password="modus" doctype=" 1" dataid="Partner_V2" filename="Partner_ApplicationPack_10DE6C09-79CF-4547-97AA-6302CF976185_0.docx" isprotected="1">
```

```

<object title="Confirmation_NewCustomer" type="B" name="Confirmation_NewCustomer"
index="02062008-759-tc-it11" reference="10DE6C09-79CF-4547-97AA-6302CF976185:7D4AD928-68CD-
4775-A471-35235B9C0B16" state="2" enabled="1" childcount="0"/>
<object title="If_you_have_more_questions" type="B" name="If_you_have_more_questio
ns" index="02062008-739-tc-it11" reference="10DE6C09-79CF-4547-97AA-6302CF976185:308DFBC5-
F520-414E-ACBD-0A4495C33301" state="2" enabled="1" childcount="2"/>
<object title="List_Accounts" type="B" name="List_Accounts" index="02062008-773-
tc-it11" reference="10DE6C09-79CF-4547-97AA-6302CF976185:D6A6A212-0B4F-4308-A835-
815678A8CC6B" state="2" enabled="1" childcount="1" />
<object title="Salutation" type="B" name="Salutation" index="02062008-776-tc-
it11" reference="10DE6C09-79CF-4547-97AA-6302CF976185:81D0CC66-8A1E-4219-BF17-
29668FFFA6B32" state="1" enabled="1" childcount="2"/>
</object>
<object title="RepCopy" type="D" name="Partner_ApplicationPack" index="02062008-824-
tcr-it11" reference="3C331B86-8552-4AD6-A242-
29106BFBCA21" state="0" enabled="1" print_asktime="0" form="Formular1" printer="" printer_t
ype="" printtime="2" print_changeable="1" copycount="0" copy="1" password="modus" copytext=
"RepCopy" originalid"10DE6C09-79CF-4547-97AA-6302CF976185:81D0CC66-8A1E-4219-BF17-
29668FFFA6B32"=/>
</object>
</process>
</mws>

```

Descrição dos atributos comuns do elemento 'object':

Atributo	Descrição
type	Tipo de objeto P = Pacote D = Modelo de documento B = Bloco de texto
name	Nome do objeto
Index	Índice do objeto
title	Título (substitui o nome exibido para usuários corporativos)
reference	Referência do objeto no pacote ModusOne (consulte: Endereço de um objeto ModusOne em um pacote)

Atributos adicionais para documento, cópias e blocos de texto

Atributo	<value>	Descrição
estado	0	Desativado (o objeto pode ser desativado, mas pode ser ativado)
	1	Obrigatório (o objeto está ativado e não pode ser desativado)

	2	Ativado (o objeto está ativado, mas pode ser desativado)
	3	Desativado (o objeto está desativado e não pode ser ativado)
ativado	0	<ul style="list-style-type: none"> para um documento: quando estado está desativado ou desabilitado para uma cópia do documento: quando o respectivo original não estiver ativado (ou seja, o original tem o estado desativado ou desabilitado)
	1	<ul style="list-style-type: none"> para um documento: quando o documento está ativado (o estado é desativado ou obrigatório) para uma cópia do documento: quando o original está ativado e a cópia tem o estado ativado ou obrigatório

Atributos adicionais para documentos/cópias

Atributo	<value>	Descrição
print_asktime	0	Consultar tempo de impressão.
	1	Não consultar tempo de impressão.
form		Nome do formulário
printer		Nome da impressora
printer_type	OEP	Impressora de email
	OP	Impressora online
printtime	0	desconhecido
	1	diretamente
	2	direcionado
print_changeable	0	Muda para definição de impressão permitida
	1	Muda para definição de impressão não permitida
copyCount		Número de cópias reais
copy	0	Original
	1	Copiar
senha		Senha do documento em texto simples
doctype	1	DynamicWord

	2	StaticWord
	3	StaticPdf
	4	StaticTiff
	5	StaticXps

O atributo a seguir só é incluído para originais:

Atributo	Descrição
dataid	Nome do TextDataReference atribuído que faz referência à seleção a ser usada.

Os atributos a seguir só são incluídos para cópias:

Atributo	Descrição
copytext	Texto na cópia
originalid	ID filho do original. O valor corresponde ao valor do atributo 'reference' do respectivo documento original

Os atributos a seguir só são incluídos se o documento ou a cópia já tiver sido criada:

Atributo	<value>	Descrição
filename		Nome do documento criado
isprotected	0	A proteção contra gravação não está ativada
	1	A proteção contra gravação está ativada

Atributos adicionais do bloco de texto:

Atributo	<value>	Descrição
childcount	0...n	Número de objetos filho no bloco de texto

O atributo a seguir só será incluído para blocos de documento/texto se um documento externo tiver sido inserido no objeto de um Cliente:

Atributo	<value>	Descrição
isexternaldocument	1	Um documento externo é atribuído

Ações incorporadas no XML do objeto

Se as propriedades do objeto específico estiverem definidas com o comando Set_Structure, as ações do pacote e do documento também poderão ser executadas.

Observação

Para documentos e blocos de texto inseridos dinamicamente, a string de referência atribuída automaticamente na entrada corresponde a um número negativo.

Descrição do atributo

Atributo	<value>	Descrição
Ação	includeDoc	Inserir um novo documento no pacote. (Documento do usuário corporativo)
	includeBlock	Inserção dinâmica do bloco de texto em um documento do pacote
	move	Mover o objeto existente (documentos do usuário corporativo e blocos de texto).
	delete	Excluir o objeto inserido (documentos do usuário corporativo e blocos de texto).
	loadexternaldocument	Carregar/remover documentos externos
target_reference		Índice de referência do objeto na árvore que determina a posição da entrada. Se esse valor não estiver especificado, o objeto será inserido no final.
position	depois de antes de	O objeto é inserido depois de target_reference.

		O objeto é inserido antes de target_reference.
name		Nome do objeto a ser inserido ou, para loadexternaldocument, o nome exclusivo do arquivo que já foi carregado ou uma string vazia para remover o documento externo novamente.
reference		Para as ações Mover/Excluir, a referência do objeto a ser movido/excluído. Para a ação loadexternaldocument, a referência do objeto para o qual o documento externo deve ser carregado/removido.
doc_reference		Referência do documento no qual o bloco de texto deve ser inserido dinamicamente.

Observação

Certifique-se que o arquivo especificado para a ação **loadexternaldocument** com o atributo **name** já foi carregado para o servidor através do método **Doc_SetFile_Mime**.

Além disso, o Cliente deve especificar um nome exclusivo para o arquivo para evitar que um arquivo existente com o mesmo nome seja substituído.

Se um arquivo externo tiver de ser removido novamente, a ação **loadexternaldocument** deve ser executada usando uma string vazia como valor para o nome do atributo.

Exemplo de uso de ações

Inserir um documento

```
<mws type="OBJECT">
  <process>
    <object action="includedoc" name="TestDocument" target_reference="" position="after"/>
  </process>
```

```
</mws>
```

Mover um documento

```
<mws type="OBJECT">
  <process>
    <object action="move" reference="-3" target_reference="-1" position="before"/>
  </process>
</mws>
```

Excluir um documento

```
<mws type="OBJECT">
  <process>
    <object action="delete" reference="-1"/>
  </process>
</mws>
```

Inserir bloco de texto (nenhum objeto de destino)

```
<mws type="OBJECT">
  <process>
    <object action="includeblock" name="DynBst1" doc_reference="-1" target_reference="-1" position="unknown" />
  </process>
</mws>
```

Inserir bloco de texto (antes do objeto de destino)

```
<mws type="OBJECT">
  <process>
    <object action="includeblock" name="DynBlock1" doc_reference="-1" target_reference="-1" position="unknown" />
  </process>
</mws>
```

Mover um bloco de texto

```
<mws type="OBJECT">
  <process>
    <object action="move" reference="-1:-2" target_reference="-1:-1" position="after"/>
  </process>
</mws>
```

```

    </process>
</mws>

```

Excluir um bloco de texto

```

<mws type="OBJECT">
  <process>
    <object action="delete" reference="-1:-2"/>
  </process>
</mws>

```

Definir documento externo para um bloco de texto

O arquivo especificado deve já ter sido carregado para o servidor pelo Cliente (método: Doc_SetFile_Mime).

```

<mws type="OBJECT">
  <process>
    <object action="loadexternaldocument" name="MyExternalBlockDocument.docx"
  " reference="9BD927E4-39CC-4217-90D1-38C3B15F4CED:35E5A161-D291-4244-84FF-6CB30F14D1A0" />
  </process>
</mws>

```

Remover um documento externo de um bloco de texto

```

<mws type="OBJECT">
  <process>
    <object action="loadexternaldocument" name="" reference="9BD927E4-39CC-4217-90D1-38C3B15F4CED:35E5A161-D291-4244-84FF-6CB30F14D1A0" />
  </process>
</mws>

```

Definição de um documento externo para um documento

O arquivo especificado deve já ter sido carregado para o servidor pelo Cliente (método: Doc_SetFile_Mime).

```

<mws type="OBJECT">
  <process>
    <object action="loadexternaldocument" name="MyExternalStaticDocument.pdf"
  " reference="C6D5FD76-516C-482E-AAA6-9E51BE8BD2E2" />
  </process>
</mws>

```

Remoção de um documento externo de um documento

```

<mws type="OBJECT">

```



```

    <process>
      <object action="loadexternaldocument" name="" reference="C6D5FD76-516C-
482E-AAA6-9E51BE8BD2E2" />
    </process>
  </mws>

```

Informações sobre XML

```

<mws type="PROCESSINFO,SELPARAMS,MANVARS,POOLVARS" version="2">
  <process id="a0deab2e-7c05-498e-8c92-0648cff1b3da" systemoid="dml">
    <title>BasicBindingTest</title>
    <mwsprocessmngnr lasterrorcode="0" lasterrormessage="" lastactionname="" onstart="">
      <mwsaction name="FORWARD" lasterrorcode="0" lasterrormessage="" state="0" configur
ed="0"/>
      <mwsaction name="PRINTANDARCHIVE" lasterrorcode="0" lasterrormessage="" state="0"
configured="0"/>
      <mwsaction name="PRINTDOCUMENTS" lasterrorcode="0" lasterrormessage="" state="0" c
onfigured="0"/>
      <mwsaction name="OBJECTSELECTION" lasterrorcode="0" lasterrormessage="" state="0"
configured="0"/>
      <mwsaction name="DATASELECTION" lasterrorcode="0" lasterrormessage="" state="0" co
nfigured="0"/>
      <selparams>
        <selections>
          <selection name="Partner_Letter" title="Partner_Letter" active="True">
            <selparam name="Partnernumber" displayname="PartnerNumberTest" mask=""
/>
          </selection>
          <selection name="Partner_V2" title="Partner_V2" active="False"/>
        </selections>
      </selparams>
    </mwsaction>
    <mwsaction name="COMPLETESTRUCTURE" lasterrorcode="0" lasterrormessage="" state="0
" configured="0"/>
    <mwsaction name="CREATEDOCUMENTS" lasterrorcode="0" lasterrormessage="" state="0"
configured="0"/>
    <manvars>
      <docref name="Partner_Application_ManVar" title="Partner_Application_ManVar"
reference="790B817A-B2C3-475E-86E5-08118150EA94">
        <manvar name="Agent_Forename" reference="DocumentCollection.790B817A-
B2C3-475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_E4437CCD-E71A-4181-9C6E-
0691B26E0C8F" title="Agent_Forename" mask="" vh_name="" vh_id="" vh_system="" input_enabled
="Y" data_link="MODUSUSER" data_attribute="Forename" input_required="0"/>
        <manvar name="Agent_Surname" reference="DocumentCollection.790B817A-B2C3-
475E-86E5-08118150EA94:308DFBC5-F520-414E-ACBD-0A4495C33301_EAAC39D9-81BA-476B-A003-
039ABF3618AA" title="Agent_Surname" mask="" vh_name="" vh_id="" vh_system="" input_enabled=
"Y" data_link="MODUSUSER" data_attribute="Surname" input_required="0"/>
      </docref>

```

```

        <docref name="Partner_Application" title="Partner_Application" reference="93
3110B7-0485-4C62-A50C-0B9D7FC62DB1"/>
    </manvars>
</mwsaction>
</mwsprocessmgr>
<pools>
  <pool name="SYSTEMPOOL" />
  <pool name="MODUSUSER">
    <poolvar name="USERID" type="S">MMueller</poolvar>
    <poolvar name="USERROLENAMES" type="S">Administrator_Role,User_Role</poolvar>
    <poolvar name="EMAIL" type="S">n.v. mail</poolvar>
    <poolvar name="TELEPHONE" type="S">n.v. telephonenumber</poolvar>
    <poolvar name="USERPRINCIPALNAME" type="S">n.v. userpricipalname</poolvar>
    <poolvar name="DISPLAYNAME" type="S">n.v. DisplayName</poolvar>
  </pool>
</pools>
</process>
</mws>

```

Logon XML

```

<?xml version="1.0" encoding="utf-16"?>
<mws>
  <loginparams>
    <loginparam name=""alias"">USERNAME</loginparam>
  </loginparams>
</mws>

```

O nome de usuário e a senha passados no logon são usados para autenticação.

Se um alias for passado, as informações do usuário (atributos/direitos do perfil) do usuário são carregadas e usadas na sessão do MWS.

XML de UserInfo

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="USERINFO" versio
n="2.0">
  <userinfo>
    <attributes>
      <attribute name="UserId" value="TestUser" />
      <attribute name="UserRoleNames" value="Administrator_Role" />
    </attributes>
    <objectpermissions>
      <objectpermission objecttype="DataProviderDefinition" objectactionpe
rmission="Create Read Update Delete" />
      <objectpermission objecttype="AliasTable" objectactionpermission="Cr
eate Read Update Delete" />
    </objectpermissions>
  </userinfo>
</mws>

```

```

    <objectpermission objecttype="TextComponent" objectactionpermission="
Create Read Update Delete" />
    <objectpermission objecttype="TextComponent" objectactionpermission="
Create Read Update Delete" />
    <objectpermission objecttype="Enclosure" objectactionpermission="Cre
ate Read Update Delete" />
    <objectpermission objecttype="TextDocumentCollection" objectactionpe
rmission="Create Read Update Delete" />
    <objectpermission objecttype="DataObject" objectactionpermission="Cr
eate Read Update Delete" />
    <objectpermission objecttype="TextDocument" objectactionpermission="
Create Read Update Delete" />
    <objectpermission objecttype="EMailPrinter" objectactionpermission="
Create Read Update Delete" />
    <objectpermission objecttype="Form" objectactionpermission="Create R
ead Update Delete" />
    <objectpermission objecttype="Envelope" objectactionpermission="Crea
te Read Update Delete" />
    <objectpermission objecttype="DataObjectQuery" objectactionpermissio
n="Create Read Update Delete" />
    <objectpermission objecttype="DataObjectScript" objectactionpermissi
on="Create Read Update Delete" />
    <objectpermission objecttype="MwsProcess" objectactionpermission="Cr
eate Read Update Delete" />
    <objectpermission objecttype="LookupTable" objectactionpermission="C
reate Read Update Delete" />
    <objectpermission objecttype="Printer" objectactionpermission="Creat
e Read Update Delete" />
    <objectpermission objecttype="Folder" objectactionpermission="Create
Read Update Delete" />
    <objectpermission objecttype="OutsourcingPrinter" objectactionpermiss
ion="Create Read Update Delete" />
    <objectpermission objecttype="PostageDefinition" objectactionpermiss
ion="Create Read Update Delete" />
    <objectpermission objecttype="Profile" objectactionpermission="Creat
e Read Update Delete" />
    <objectpermission objecttype="Process" objectactionpermission="Creat
e Read Update Delete" />
    <objectpermission objecttype="Role" objectactionpermission="Create R
ead Update Delete" />
    <objectpermission objecttype="RoleMapping" objectactionpermission="C
reate Read Update Delete" />
    <objectpermission objecttype="Selection" objectactionpermission="Cre
ate Read Update Delete" />
    <objectpermission objecttype="Script" objectactionpermission="Create
Read Update Delete" />
    <objectpermission objecttype="StandardProcess" objectactionpermissio
n="Create Read Update Delete" />

```

```

    <objectpermission objecttype="SimpleTextContainer" objectactionpermission="Create Read Update Delete" />
    <objectpermission objecttype="TransferJob" objectactionpermission="Create Read Update Delete" />
    <objectpermission objecttype="ValueHelp" objectactionpermission="Create Read Update Delete" />
    <objectpermission objecttype="CustomUI" objectactionpermission="Create Read Update Delete" />
    <objectpermission objecttype="XsdSelection" objectactionpermission="Create Read Update Delete" />
  </objectpermissions>
  <permissions>
    <permission>Administrate</permission>
    <permission>CommonSettings</permission>
    <permission>MwsInsertBlock</permission>
    <permission>MwsClearCache</permission>
    <permission>MwsDataSelection</permission>
    <permission>MwsInsertDocument</permission>
    <permission>MwsCreateDocuments</permission>
    <permission>CanPrintInPreview</permission>
    <permission>MwsPrintAndArchive</permission>
    <permission>MwsSetBundlePrinter</permission>
    <permission>MwsSetDocumentPrinter</permission>
    <permission>MwsSetDocumentPrinterWizzard</permission>
    <permission>MwsDownloadDocument</permission>
    <permission>MwsLoadExternalBlockFile</permission>
    <permission>MwsLoadExternalDocumentFile</permission>
    <permission>MwsAdministrate</permission>
    <permission>MwsViewOpenEnvelope</permission>
    <permission>MwsViewProcess</permission>
    <permission>MwsViewStack</permission>
    <permission>MwsViewOpenJob</permission>
    <permission>MwsShowNavigator</permission>
    <permission>MwsChangeFolder</permission>
    <permission>MwsChangeSystem</permission>
    <permission>ExecuteModusStudio</permission>
    <permission>ShowNavigator</permission>
    <permission>ShowTypeFilter</permission>
    <permission>OdinJobRemove</permission>
    <permission>OdinJobSetPrinter</permission>
    <permission>OdinJobSetState</permission>
    <permission>OdinStackEnvelopeRemove</permission>
    <permission>OdinProcessSuspend</permission>
    <permission>OdinProcessUnlock</permission>
    <permission>OdinProcessDelete</permission>
    <permission>OdinProcessSetBack</permission>
  </permissions>

```

```

<permission>OdinStackSetBackToStreaming</permission>
<permission>OdinStackEdit</permission>
<permission>OdinStackUnlock</permission>
<permission>OdinStackFree</permission>
<permission>OdinStackDelete</permission>
<permission>OdinStackLock</permission>
<permission>OdinOpenJobCreateStack</permission>
<permission>OdinProcessCreateStack</permission>
<permission>OdinStackSetBack</permission>
<permission>OdinStackSetBackTo</permission>
<permission>ChangeFolder</permission>
<permission>MwsManualForwardProcess</permission>
<permission>MwsProcessList</permission>
<permission>MwsProcessListFromDefinedUser</permission>
<permission>ModifyFontFace</permission>
<permission>ModifyFontColour</permission>
<permission>ModifyFontStyleBold</permission>
<permission>ModifyFontStyleItalic</permission>
<permission>ModifyFontStyleUnderline</permission>
<permission>ModifyFontSize</permission>
<permission>Search</permission>
<permission>ChangeSystem</permission>
<permission>SynchronizeSystem</permission>
<permission>TransferSystem</permission>
<permission>Test</permission>
<permission>ModifyFontAlignment</permission>
<permission>MwsForwardedProcessList</permission>
</permissions>
</userinfo>
</mws>

```

Erro XML

```

<?xml version="1.0" encoding="utf-16"?>
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="ERROR" version="
2.0">
  <error location="functionname" errorcode="numeric Code">error message</err
or>
</mws>

```

Funções

Limpar cache

Limpa o cache do RepositoryRuntimeService.

Sintaxe

ClearCache(string sessionId, out string xmlResult)

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
xmlResult	Vazio quando ok, caso contrário, MwsError

Retorna o valor

0	OK
<>0	Erro

Doc_GetFile_Mime

Permite o download de um documento já criado no servidor. O documento é retornado como uma string com código base64.

Sintaxe

```
Doc_GetFile_Mime(string sessionId, string processId,
                 string fileName, out string doc,
                 out string xmlResult )
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo cujo objeto de negócios deve ser gerenciado.
fileName	Nome do arquivo a ser retornado.
doc	String com código Base64, representado o conteúdo do arquivo.
xmlResult	Vazio quando ok

Retorna o valor

0	OK
<>0	Erro

Doc_SetFile_Mime

Faz upload de um arquivo (string com código base64) para o servidor.

Sintaxe

```
Doc_SetFile_Mime(string sessionId, string processId,
                string fileName, string doc,
                out string xmlResult
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo cujo objeto de negócios deve ser gerenciado.
fileName	Nome do arquivo cujo conteúdo deverá ser gravado.
doc	String com código Base64, representado o conteúdo do arquivo.
xmlResult	Vazio quando ok

Retorna o valor

0	OK
<>0	Erro

Logon

Esta função é usada para efetuar logon no servidor. A função de logon deve ser executada uma vez ao iniciar uma sessão do servidor e é válida pela duração da sessão.

O ID da sessão (Identificador da sessão) retornado deve ser usado para todas as chamadas subsequentes.

Sintaxe

```
Login(string userName, string password, string code,
       string options, out string sessionID,
       out string xmlResult )
```

Parâmetro

userName	Nome de usuário
senha	Senha em texto simples. Usada somente quando o parâmetro 'code' contém uma string vazia

code	Senha criptografada (criada previamente usando Encoder.exe)
options	Logon XML, consulte: Logon XML
sessionID	Retorna o ID da sessão a ser usado para chamadas subsequentes
xmlResult	O respectivo XML de UserInfo Xml quando a execução do logon tiver êxito, caso contrário, Erro de XML do MWS

Retorna o valor

0	OK
<>0	Erro

Logoff

Esta função é usada para fazer logoff da sessão. Observe que uma sessão deve sempre ser finalizada com um logoff para excluí-la do servidor. Se essa regra não for observada, os conjuntos de dados estão à esquerda na respectiva tabela.

Sintaxe

Logout(string sessionId, out string xmlResult)

Parâmetro

sessionId	ID da sessão válida da sessão a ser encerrada.
xmlResult	Vazio quando ok, caso contrário, MwsError

Retorna o valor

0	OK
<>0	Erro

Obj_GetStructure

Retorna a estrutura do objeto referenciado.

Sintaxe

int Obj_GetStructure(string sessionId, string processId,


```
string rootRef,int maxLevel,
out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)	
processId	ID do processo do objeto que está sendo referenciado.	
rootRef	Referência do objeto dentro do processo	
maxLevel	-1	tudo
	0	só o objeto
	<> 0	a estrutura incluindo todos os elementos filho no nível n-th
xmlResult	XML do objeto dependendo do componente	

Retorna o valor

0	OK
<>0	Erro

Obj_SetStructure

Permite definir propriedades de objetos já existente ou usar comandos para executar ações do pacote ou documento. Para obter mais informações, consulte o capítulo sobre Ações incorporadas no XML do objeto.

Sintaxe

```
Obj_SetStructure(string sessionId,string processId,
string content,string rootRef,int maxLevel,
out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo do objeto que está sendo referenciado.
content	As informações do objeto de negócios a ser definido.
rootRef	Referência do objeto cuja estrutura deve ser retornada após a chamada

maxLevel	Determina quantos níveis devem ser retornados (-1 = todos)
xmlResult	Se estiver ok, a estrutura do objeto de negócios

Retorna o valor

0	OK
<>0	Erro

Obj_Toggle

Alterna o status de um objeto e retorna a estrutura modificada. Observe que alternar objetos com base em grupos e ações pode levar a outros objetos sendo alternados automaticamente.

A função retorna a estrutura modificada depois que todas as ações de alternância respectivas foram executadas.

Sintaxe

```
Obj_Toggle(string sessionId,string processId,string objRef,
           string rootRef,int maxLevel,out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo do objeto que está sendo referenciado.
objRef	Referência do objeto a ser alternado.
rootRef	Referência do objeto cuja estrutura deve ser retornada após a chamada
maxLevel	Determina quantos níveis devem ser retornados (-1 = todos)
xmlResult	A estrutura do objeto, se tiver êxito.

Retorna o valor

0	OK
<>0	Erro

Ping

Retorna uma mensagem padrão para fins de teste.

Sintaxe

Ping()

Parâmetro

-

Retorna o valor

Mensagem padrão **Serviço do processo MWS**.

Process_Close

Fecha um processo com o passado em status.

Sintaxe

Process_Close(string sessionId,string processId,int status,string options,out string xmlResult)

Parâmetro

sessionId	Identificador de sessão válida (SessionId)	
processId	ID do processo a ser fechado.	
status	Status a ser definido:	
	0	<p>Salvar processo</p> <p>A manipulação de processos com status 0 pode ser reiniciada.</p>
	1	<p>O processo está sendo editado no momento.</p> <p>O valor 1 é usado internamente e não deve ser passado, pois isso causa um erro.</p>
	2	<p>O processo foi concluído com êxito.</p> <p>O serviço MWS_Assistant exclui automaticamente processos salvos com o status 2.</p>
3	<p>O processo estava com falha, por exemplo, anulado pelo usuário.</p> <p>Processos com defeito ou falhas não são excluídos automaticamente pelo serviço MWS_Assistant.</p> <p>O ModusOne Client exclui explicitamente os processos anulados por um usuário.</p>	

	Outros valores	O processo está fechado.
options	Opções XML para fechar um processo com especificação do título e descrição. Exemplo de XML para configurar título e descrição: <?xml version="1.0" encoding="utf-16"?> <mws> <process> <title>My Title</title> <description>My description</description> </process> </mws>	
xmlResult	Vazio quando ok	

Retorna o valor

0	OK
<>0	Erro

Process_Create

Cria um novo processo MWS.

Para criar um novo processo, as seguintes ações são executadas no servidor:

- Carregue os processos do ModusOne que definem a sequência de etapas do processo MWS do repositório (se já não estiver feito).
- Crie um diretório para os dados do respectivo processo de criação de documentos.
- Crie um conjunto de dados para os processos na tabela de banco de dados MWS_Processes.
- Avalie as opções se uma Opção XML foi passada (parâmetro **options**).
- Execute **OnSetOptionsScript** do processo MWS, se atribuído.
- Execute o script **OnNewProcess** do processo MWS, se atribuído.
- Execute um comando de início se passado para as Opções XML.

Se ocorrer um erro durante a execução deste método, os dados do processo criado (arquivos e conjunto de dados BD) serão excluídos.

Sintaxe

Process_Create(string sessionId,ref string processId,

```
string mSystem,string mType,string mTitle,
string objectIndex,string objectName,
string objectType,string objectData,
string options,out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	Se o ID do processo não for passado, um novo será criado.
mSystem	A OID de sistema do objeto a ser carregado reside em.
mType	Texto livre para definir o tipo de processo.
mTitle	Texto livre para definir um título para o processo.
objectIndex	Índice do objeto a ser carregado (se este parâmetro for usado, o nome será ignorado).
objectName	Nome do objeto a ser carregado (será ignorado se o índice for especificado).
objectType	Tipo de objeto a ser carregado (no momento, somente P = pacote)
objectData	Dados opcionais de XML para um pacote. Os dados são atribuídos à primeira seleção passiva no script OnNewStart.
options	Opções XML para o processo MWS.
xmlResult	Se o processo foi criado com êxito, a estrutura do pacote ou as informações do processo são retornadas. A estrutura do pacote é retornada se um comando de início é passado às opções.

Retorna o valor

0	OK
<>0	Erro

Process_Delete

Exclui um processo aberto e libera seus recursos do servidor.

Sintaxe

```
Process_Delete(string sessionId,string processId,
```

```
out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo a ser excluído.
xmlResult	Vazio quando OK, caso contrário, MwsError

Retorna o valor

0	OK
<>0	Erro

Process_Forward

Encaminha um processo.

Sintaxe

```
Process_Forward(string sessionId,string processId,
                string destUser,string destUsergroup,
                string options,out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo a ser encaminhado.
destUser	Nome do usuário de destino para o qual o processo será encaminhado.
destUsergroup	Nome do grupo de destino para o qual o processo será encaminhado.
options	Fechar opções XML... título e descrição
xmlResult	Vazio quando ok

Retorna o valor

0	OK
<>0	Erro

Process_GetInfo

Retorna informações sobre o processo.

Sintaxe

```
int Process_GetInfo(string sessionId,string processId,
                   string infoType,string options,
                   out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo para o qual as informações serão retornadas.
infoType	String separada por vírgulas contendo as informações a serem recuperadas. Em primeiro lugar, essas informações são de tipos predefinidos: PROCESSINFO POOLVARS e, em segundo lugar, dependentes do processo MWS e são carregadas e avaliadas pelo script OnGetInfo.
options	Não suportado no momento.
xmlResult	ProcessInfo XML, se OK

Retorna o valor

0	OK
<>0	Erro

Process_GetLastError

Retorna o último erro de um processo.

Sintaxe

```
Process_GetLastError(string sessionId,string processId,
                    out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo para o qual o último erro será recuperado.

xmlResult	Se tiver êxito, retornará o último erro XML.
-----------	--

Retorna o valor

0	OK
<>0	Erro

Process_GetList

Retorna uma lista de todos os processo existentes no momento.

Sintaxe

```
Process_GetList(string sessionId,int startAt,int max,
                string options,out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
startAt	Iniciar índice
max	Número máximo de processo a serem retornados.
options	<p>Este parâmetro pode ser usado para especificar quais processos incluir na lista de processos recuperados.</p> <p>As configurações possíveis são:</p> <ol style="list-style-type: none"> 1. Retorna os processos criados pelo usuário (padrão quando string vazia é passada). 2. Retorna os processos encaminhados ao usuário diretamente ou a um grupo ao qual o usuário pertence. 3. Retorna processos próprios e processos encaminhados. Se uma string vazia for passada para este parâmetro, somente os processos próprios de um usuário serão retornados (consulte 1. acima).
xmlResult	Um XML MwsProcessList quando ok, caso contrário, MwsError.

Retorna o valor

0	OK
<>0	Erro

Process_Open

Reinicia o processamento de um processo e define o nome do usuário como o do usuário atual.

Sintaxe

```
Process_Open(string sessionId,string processId,string options,
             out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo a ser aberto.
options	Não usado no momento
xmlResult	Se estiver OK, a estrutura do objeto do objeto de negócios

Retorna o valor

0	OK
<>0	Erro

Process_SetData

Define dados XML para o objeto de negócios.

Sintaxe

```
Process_SetData(string sessionId,string processId,string dataType,
                string data,out string xmlResult)
```

Parâmetro

sessionId	Identificador de sessão válida (SessionId)
processId	ID do processo para o qual os dados serão definidos.
dataType	Nome a ser atribuído aos dados. No momento, refere-se ao nome da referência de dados.
data	Os dados XML a serem definidos.
xmlResult	Vazio quando ok

Retorna o valor

0	OK
<>0	Erro

Process_Start

Inicia uma ação do processo.

Sintaxe

```
Process_Start(string sessionId,string processId,string command,
              string options,out string xmlResult)
```

Parâmetro

ses sio nId	Identificador de sessão válida (SessionId)
pro ces sId	ID do processo onde a ação será iniciada.
co mm and	Este parâmetro contém o comando a ser passado. Os nomes dos comandos são derivados do script das Atividades da chamada do processo MWS.
opti ons	Este parâmetro pode ser usado para definir várias opções do processo. Ele pode ser usado para definir atividades do processo, passar variáveis etc.

xml
Res
ult

A estrutura do pacote, se tiver êxito.

Exemplo

```
<?xml version="1.0" encoding="utf-16"?>
<mws type="OBJECT">
  <process id="b9187257-812a-4e88-afa6-facb621dd2eb">
    <object title="ApplicationPack" type="P" name="ApplicationPack" index="02062008-822-tdc-itl1" reference="">
      <object title="Partner_ApplicationPack" type="D" name="Partner_ApplicationPack" index="02062008-823-tdr-itl1" reference="10DE6C09-79CF-4547-97AA-6302CF976185" state="1" enabled="1" print_asktime="0" form="DIN_A4_80g/qm" printer="" printer_type="" printtime="2" print_changeable="0" copycount="0" copy="0" password="modus" filename="Partner_ApplicationPack_10DE6C09-79CF-4547-97AA-6302CF976185_O.docx" />
      <object title="pagenum" type="D" name="pagenum" index="23062009-591-tdr-
```

```
dm" reference="A7DE4DF9-00A4-41FC-8A84-
05BB1D9D893D" state="2" enabled="1" print_asktime="0" form="DIN_A4_80g/qm" printer=""
printer_type="" printtime="2" print_changeable="1" copycount="0" copy="0" password=""
" filename="pagenum_A7DE4DF9-00A4-41FC-8A84-05BB1D9D893D_O.docx" />
    <object title="RepCopy" type="D" name="Partner_ApplicationPack" index="02062
008-824-tcr-itl1" reference="3C331B86-8552-4AD6-A242-
29106BFBCA21" state="2" enabled="1" print_asktime="0" form="DIN_A4_80g/qm" printer=""
printer_type="" printtime="2" print_changeable="0" copycount="0" copy="1" password=""
modus" copytext="Copy for Representative" />
    <object title="AgentCopy" type="D" name="Partner_ApplicationPack" index="020
62008-825-tcr-itl1" reference="2572D52B-D2EC-4864-81A3-
A4017CD16F53" state="2" enabled="1" print_asktime="0" form="DIN_A4_80g/qm" printer="F
inePrint" printer_type="OP" printtime="1" print_changeable="1" copycount="0" copy="1"
password="modus" copytext="Copy for Agent" />
    </object>
</process>
</mws>
```

Retorna o valor

0	OK
<>0	Erro

Repositório do ModusOne WS

Exemplos de XML

XML de Systemdef

```
<?xml version="1.0" encoding="UTF-8"?>
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="SYSTEMDEF" version="2.0">
    <systemdef name="Odin_Standard" description="" systemoid="odv"/>
</mws>
```

MwsItemInfo XML

```
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="ITEMINFO" versio
n="2.0">
    <iteminfo>
        <IsProtected>>false</IsProtected>
        <IsReadOnly>>false</IsReadOnly>
        <ObjectName>Partner_Application</ObjectName>
        <ObjectId>02062008-805-td-itl1</ObjectId>
        <DatabaseId>294</DatabaseId>
        <ObjectType>TextDocument</ObjectType>
```

```

    <LastModifiedAt>2010-09-01T11:28:26,617</LastModifiedAt>
    <LastModifiedBy>MEGATRON\Administrator</LastModifiedBy>
    <CreatedAt>2008-06-02T13:05:51.937</CreatedAt>
    <CreatedBy>DT_MMUELLER\MMueller</CreatedBy>
    <LockedBy />
    <LockedAt xsi:nil="true" />
    <Context>None</Context>
  </iteminfo>
</mws>

```

XML MwsSystemList

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="SYSTEMLIST" version="2.0">
  <systems>
    <system name="MWS_Standard" description="" systemoid="mws"/>
    <system name="ModBatch_Standard" description="" systemoid="modb"/>
    <system name="User_Standard" description="" systemoid="user"/>
    <system name="Odin_Standard" description="" systemoid="odv"/>
    <system name="TestSystem" description="" systemoid="ts"/>
    <system name="ModusTest" description="" systemoid="modtest"/>
  </systems>
</mws>

```

XML MwsNavigTree

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="NAVIGTREE"
version="2.0" systemname="it2">
  <folder name="ImportTest_2" type="A" path="/" dbid="0">
    <folders>
      <folder name="Test1" type="A" path="/Test1/" dbid="126">
        <folders>
          <folder name="Test1_1" type="A" path="/Test1/Test1_1/" dbid="2
371"/>
        </folders>
      </folder>
      <folder name="Test2" type="A" path="/Test2/" dbid="897"/>
    </folders>
  </folder>
</mws>

```

MwsNavigList XML

Este XML é usado para navegar pelo repositório do ModusOne.

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="NAVIGLIST" version="2.0">
  <foldercontent path="" filter="A,P,D,B" systemname="it12">

```

```

<items>
  <item name="Paragraph1" objectid="14102009-1768-tc-itl2"
  type="B" />
  <item name="Paragraph2" objectid="14102009-1769-tc-itl2"
  type="B" />
  <item name="KeepParagraphsTogether" objectid=
  "14102009-1767-tc-itl2" type="B" />
  <item name="KeepParagraphsTogetherTables"
  objectid="14102009-1770-tc-itl2" type="B" />
  <item name="Sender_Date" objectid="14102009-1759-tc-itl2"
  type="B" />
  <item name="Partner_Application" objectid="14102009-1790-td-itl2"
  type="D" />
  <item name="Partner_Application_XDOC" objectid="14102009-1799-td-
itl2"
  type="D" />
  <item name="XML-Testdocument" objectid="14102009-1798-td-
itl2" type="D" />
  <item name="Application" objectid="14102009-1807-tdc-
itl2" type="P" />
  <item name="Application_XDOC" objectid="14102009-1812-tdc-
itl2" type="P" />
  <items>
</foldercontent>
</mws>

```

Atributos do elemento foldercontent

parentpath	Caminho principal
path	Caminho para o qual os objetos serão listados
filtro	Filtro usado
user	ID do usuário atual
systemname	Nome do sistema

Atributos do elemento item

type	Tipo de objeto
name	Nome do objeto

MwsFolderContent XML

```
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="FOLDERCONTENT" v
ersion="2.0">
<foldercontent systemname="it2">
  <items>
    <item name="Test1_1" objectid="19102009-1230-f-it2" type="A"/>
    <item name="TextBlock" objectid="it2-tc-17" type="B"/>
    <item name="TextBlockWithVariable" objectid="it2-tc-18" type="B"/>
  </items>
</foldercontent>
</mws>
```

MwsLookupResult XML

Para Rep_GetLookupValue (contém MwsLookupValue xml):

```
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="LOOKUPRES" versi
on="2.0" reference="VariablenVorgabe|15042008-666-lt-
it2|447|LookupTable|None|dm1">
  <lookup aliastable="DefaultVariables" alias="Var1">Wert1|"\"(999 99\) 99999
;1;_"</lookup>
</mws>
```

Para Rep_GetLookupObjects (contém MwsFolderContent xml):

```
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="LOOKUPRES" versi
on="2.0" reference="SystemAliasTable|15042008-665-at-
it2|446|AliasTable|None|dm1">
  <foldercontent systemname="dm1">
    <items>
      <item name="DefaulVariables" objectid="15042008-666-lt-
it2" type=""/>
    </items>
  </foldercontent>
</mws>
```

MwsDataProviderDefinition XML

```
<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="DATAPROVIDERDEF" version="2.0"
>
  <providerxml>&lt;?xml version="1.0" encoding="utf-
16"?&gt;&lt;ms:dataprowiderdefinitiontree xmlns:ms="http://www.modus-
suite.de/2006/XMLSchema"&gt;&lt;dataprowiderdefinition objectid="09072009-28-dpd-
mws" objectname="All_SQL_MWS_Sessions" description="" xml="" category="" rowsperpage="50" t
ables="MWS_SESSION" dbalias="MWS|21012008-1-dba-
mws|2|DbAlias|DatabaseAlias|mws" protected="False"&gt;&lt;sqlcolumns objectname="SqlColumnC
```

```

ollection" description=""&gt;&lt;sqlcolumn objectname="MWSS_SESSION_ID" description="" titl
e="" dbcolumnname="MWSS_SESSION_ID" size="-
1" isvisible="True" /&gt;&lt;sqlcolumn objectname="MWSS_CREATED" description="" title="" db
columnname="MWSS_CREATED" size="-
1" isvisible="True" /&gt;&lt;sqlcolumn objectname="MWSS_USERNAME" description="" title="" d
bcolumnname="MWSS_USERNAME" size="-
1" isvisible="True" /&gt;&lt;sqlcolumn objectname="MWSS_MODUSUSERINFO" description="" title
="" dbcolumnname="MWSS_MODUSUSERINFO" size="-
1" isvisible="True" /&gt;&lt;/sqlcolumns&gt;&lt;/dataproviderdefinition&gt;&lt;/ms:datapro
viderdefinitiontree&gt;
  </providerxml>
</mws>

```

XML MwsValueHelpDefinition

```

<mws xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" type="VALUEHELP" versi
on="2.0" valuehelptype="3" reference="Date_and_Time|27102008-182-vhl-
dm|15|ValueHelp|None|odv">
  <definition>&lt;datetimepicker xmlns:xsi="http://www.w3.org/2001/XMLSchema
-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema customformat="dd.MM.yyyy
HH:mm:ss" format="Custom" /&gt;
  </definition>
</mws>

```

Opções XML de MwsNavigTree

```

<mws>
  <navigtree>
    <item name="startfoldername" value="[OrdnerName]" />
  </navigtree>
</mws>

```

A definição do nome de uma pasta especifica a pasta da qual iniciar a listagem de uma hierarquia de árvore/pasta do diretório.

Se uma string for passada, a especificação é ignorada, ou seja, a hierarquia completa, começando da pasta raiz do sistema, será retornada.

Se a pasta especificada não existir no sistema, um Erro XML correspondente será retornado.

Funções

Ping

Retorna uma mensagem padrão para fins de teste.

Sintaxe

Ping()

Parâmetro

-

Retorna o valor

Mensagem padrão: **Serviço do repositório do MWS**

Rep_GetBinFile_Mime

Retorna os dados BLOB de um objeto.

Sintaxe

```
Rep_GetBinFile_Mime(string systemName, string options,
                    string objectId, string objectName,
                    string objectType, out string doc,
                    out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
options	O parâmetro é ignorado no momento e deve ser passado como uma string vazia.
objectId	ID do objeto cujos dados BLOB deverão ser recuperados. Se não estiverem definidos, os parâmetros ObjectName e ObjectType deverão ser definidos.
objectName	Nome do objeto. Se especificado, o tipo também deverá ser especificado.
objectType	O tipo dos dados do objeto a ser recuperado (por exemplo, D, B)
doc	O conteúdo do campo BLOB como string com código Base64.
xmlResult	Vazio quando OK, caso contrário, Erro de XML do MWS

Retorna o valor

0	OK
<>0	Erro

Rep_GetDataProviderDefinition

Lê um DataProviderDefinition do repositório.

Sintaxe

```
Rep_GetDataProviderDefinition(
    string systemName,string dataProviderDefinitionId,
    string dataProviderDefinitionName,
    out string xmlResult)
```

Parâmetro

systemName	SystemOld
dataProviderDefinitionId	ID do objeto de DataproviderDefinition. Se não estiver definido, o nome de DataProviderDefinition deverá ser definido
dataProviderDefinitionName	Nome de DataProviderDefinition
xmlResult	XML serializado do objeto MwsDataProviderDefinition

Retorna o valor

0	OK
<>0	Erro

Rep_GetFolderContent

Retorna o conteúdo de uma pasta.

Sintaxe

```
Rep_GetFolderContent(string systemName,string options,
    string filter,string folderName,
    out string xmlResult)
```

Parâmetro

systemName	SystemOld of the system
options	Não usado no momento
filtro	Configurações de filtro da consulta (D,P,B,A)
folderName	DBID da pasta ou nome da pasta
xmlResult	MwsFolderContent XML serializado ou MwsError no caso de um erro

Retorna o valor

0	OK
<>0	Erro

Rep_GetForms

Retorna uma lista de formulários do repositório do ModusOne.

Se o nome ou o ID do objeto de uma impressora estiver especificado, somente os formulários usados pela impressora serão listados.

Se o ID do objeto da impressora estiver especificado, o parâmetro **PrinterName** será ignorado.

Sintaxe

```
Rep_GetForms(string systemName,string usedByPrinterId,
             string usedByPrinterName,string printerType,
             out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
usedByPrinterID	Opcional: PrinterOID
usedByPrinterName	Opcional: PrinterName só é usado quando PrinterOID não está definido
printerType	Opcional: obrigatório se usedByPrinterID estiver definido. Os valores possíveis são "OEP" para impressora de email, "OP" para impressora online
xmlResult	Lista de formulários serializada como MwsFolderContent XML ou MwsError no caso de um erro

Retorna o valor

0	OK
<>0	Erro

Rep_GetItemInfo

Lê um ItemInfo do repositório.

Sintaxe

```
Rep_GetItemInfo(string systemName, string objectId,
                string objectName, string objectType,
                out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
objectId	ID do objeto para o qual o ItemInfo será lido. Se este parâmetro não estiver definido, os parâmetros ObjectName e ObjectType deverão ser definidos.
objectName	Nome do objeto para o qual ItemInfo será lido. Se o parâmetro ObjectName estiver definido, ObjectType também deverá ser definido.
objectType	Tipo do objeto para o qual ItemInfo será lido.
xmlResult	XML serializado do objeto MwsItemInfo ou MwsError no caso de um erro. Vazio se o objeto especificado não for encontrado.

Retorna o valor

0	OK
<>0	Erro

Rep_GetLookupObjects

Pesquisa os valores para um alias em uma tabela de alias e interpreta os resultados como referências de objeto (RepositoryReference(s)).

Sintaxe

```
Rep_GetLookupObjects(string systemName, string options,
                     string aliasTableId, string aliasTableName,
                     string alias, out string xmlResult)
```

Parâmetro

systemName	SystemOld
options	Não avaliado no momento

aliasTableId	ID do objeto da tabela de alias. Se o ID do objeto não estiver definido, o nome e o tipo do objeto deverão ser definidos
aliasTableName	Nome da tabela de alias na qual a pesquisa será executada.
alias	Alias cujos valores serão lidos
xmlResult	Retorna as referências de objeto na forma de um MwsLookupResult Xml (contém MwsFolderContent Xml)

Retorna o valor

0	OK
<>0	Erro

Rep_GetLookupValue

Retorna o valor de uma chave especificada da tabela de pesquisa definida.

Sintaxe

```
Rep_GetLookupValue(string systemName,string options,
                    string lookupTableId,string lookupTableName,
                    string key,out string xmlResult)
```

Parâmetro

systemName	SystemOld
options	Não avaliado no momento
lookupTableId	ID do objeto da tabela de pesquisa. Se não estiverem definidos, o nome e o tipo do objeto deverão ser definidos
lookupTableName	Nome da tabela de pesquisa
key	Chave
xmlResult	Retorna as referências de objeto na forma de um MwsLookupResult Xml (contém MwsLookupValue Xml)

Retorna o valor

0	OK
---	----

<>0 Erro

Rep_GetNavigList

Retorna uma lista de objetos em uma pasta em um sistema do ModusOne. Esta função requer a definição de um caminho e considera uma pasta de início definida para um usuário.

Internamente, isso é anexado automaticamente ao início da definição do caminho.

Sintaxe

```
Rep_GetNavigList(string systemName,string options,
                 string filter,string path,
                 out string xmlResult)
```

Parâmetro

systemName	SystemOld
options	O parâmetro é ignorado no momento e deve ser passado como uma string vazia.
filtro	Lista separada por vírgulas com identificadores de tipo dos objetos ModusOne contidos na lista de resultados. Exemplo: "A,P" (pasta e pacote)
path	ID do BD da pasta ou a definição de um caminho totalmente qualificado (/Name/Name1/Name2)
xmlResult	Um MwsNavigList Xml quando a execução tiver êxito, caso contrário, Erro de XML do MWS

Retorna o valor

0	OK
<>0	Erro

Rep_GetNavigTree

Retorna a árvore do navegador de um sistema do ModusOne.

Sintaxe

```
Rep_GetNavigTree(string systemName,string options,
                 out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
options	Opções XML, consulte: Opções XML de MwsNavigTree
xmlResult	MwsNavigTree-XML quando OK, caso contrário, Erro de XML do MWS

Retorna o valor

0	OK
<>0	Erro

Rep_GetPrinters

Retorna uma lista de impressoras online e impressora de email do repositório do ModusOne.

Se o nome ou o ID do objeto de um formulário estiver especificado, somente as impressoras usando esse formulário serão listadas.

Se o ID do objeto do formulário estiver especificado, o parâmetro **FormName** será ignorado.

Sintaxe

```
Rep_GetPrinters(string systemName, string containingFormId,
                string containingFormName,
                out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
containingFormId	Opcional: ID do objeto do formulário
containingFormName	Opcional: Nome do formulário (usado somente quando o ID do formulário não está especificado)
xmlResult	Lista de impressoras serializada como MwsFolderContent XML ou MwsError no caso de um erro

Retorna o valor

0	OK
<>0	Erro

Rep_GetSystem

Retorna informações do sistema do repositório do ModusOne.

Sintaxe

```
Rep_GetSystem(string systemName,out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema para o qual as informações serão retornadas
xmlResult	SystemDef XML quando OK, caso contrário, Erro de XML do MWS

Retorna o valor

0	OK
<>0	Erro

Rep_GetSystemList

Retorna a lista do sistema.

Sintaxe

```
Rep_GetSystemList(string options,out string xmlResult)
```

Parâmetro

options	Não usado no momento
xmlResult	MwsSystemList xml

Retorna o valor

0	OK
<>0	Erro

Rep_GetValueHelpDefinition

Lê uma ajuda de valor do repositório.

Sintaxe

```
Rep_GetValueHelpDefinition(string systemName,
                           string valueHelpId,
                           string valueHelpName,
                           out string xmlResult)
```

Parâmetro

systemName	SystemOld do sistema
valueHelpId	ID do objeto da ajuda de valor. Se não estiver definido, o nome da ajuda de valor deverá ser definido.
valueHelpName	Nome da ajuda de valor
xmlResult	XML serializado do objeto MwsValueHelpDefinition

Retorna o valor

0	OK
<>0	Erro

ModusOne WS UserRepository

Exemplos de XML

RoleList XML

```
<GetRolesResult xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/
Arrays" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <a:string>Admin_Role</a:string>
  <a:string>Administrator_Role</a:string>
  <a:string>User_Role</a:string>
  <a:string>MainUser_Role</a:string>
  <a:string>ProdDev_Role</a:string>
  <a:string>BusinessUser_Role</a:string>
  <a:string>TextAdmin_Role</a:string>
  <a:string>OtherUser_Role</a:string>
</GetRolesResult>
```

XML de UserList para GetUsers

```
<GetUsersResult xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays" xmlns:i="http://
www.w3.org/2001/XMLSchema-instance">
  <a:string>DepartmentHead</a:string>
  <a:string>ACTUser</a:string>
```



```

<a:string>Admin</a:string>
<a:string>Administrator</a:string>
<a:string>ASPNET</a:string>
<a:string>Mike</a:string>
<a:string>MMueller</a:string>
<a:string>BusinessUser</a:string>
<a:string>SQLDebugger</a:string>
</GetUsersResult>

```

XML de UserList para GetUsersOfRole

```

<GetUsersOfRoleResult xmlns:a="http://schemas.microsoft.com/2003/10/Serializ
ation/Arrays" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <a:string>Admin</a:string>
  <a:string>Administrator</a:string>
  <a:string>ASPNET</a:string>
  <a:string>Mike</a:string>
  <a:string>MMueller</a:string>
</GetUsersOfRoleResult>

```

Funções

GetRoles

Retorna todas a funções de Modus do RoleMapper configurado.

Sintaxe

```
GetRoles()
```

Parâmetro

-

Retorna o valor

Matriz de string de todas as funções de Modus existentes (RoleList XML)

GetUsersOfRole

Retorna todos os usuários que pertencem à função de Modus especificada.

Sintaxe

```
GetUsersOfRole(string roleName)
```

Parâmetro

roleName	Nome da função para a qual os usuários serão retornados.
----------	--

Retorna o valor

Matriz de string de todos os nomes de usuários que pertencem à função de Modus especificada (UserList XML).

GetUsers

Retorna todos os usuários que pertencem a qualquer função de Modus no RoleMapper.

Sintaxe

`GetUsers()`

Parâmetro

-

Retorna o valor

Matriz de string de todos os nomes de usuários que pertencem a qualquer função de Modus no RoleMapper (UserList XML)

Ping

Retorna uma mensagem padrão para fins de teste.

Sintaxe

`Ping()`

Parâmetro

-

Retorna o valor

Mensagem padrão: **Serviço do Modus UserRepository**

Valores de retorno

Valores de retorno de funções do MWS

Como regra, todas as funções do servidor MWS retornam um número.

Esse número especifica se a função foi executada com êxito.

Basicamente, para um código <> 0 no XML, um erro XML é retornado. Esse XML descreve o erro em mais detalhes.

Os códigos de resultado podem ser agrupados em três categorias:

- Resultado da execução com êxito
- Códigos de erro
- Avisos

Resultado da execução com êxito

0	Chamada processada com êxito.
---	-------------------------------

Códigos de erro

-1	O erro não pode ser identificado exatamente, detalhes na mensagem de erro XML
1	Erro durante a chamada da função, detalhes na mensagem de erro XML
302	Erro ao definir as opções passadas
303	Erro ao definir a estrutura do pacote
304	Erro ao fechar o pacote
305	Erro ao iniciar recuperação de dados
309	Número máximo de licenças de clientes excedidas. Esse código de retorno é gerado quando: <ul style="list-style-type: none"> • O número máximo de clientes definido na licença do servidor excede o número de sessões na tabela de sessão. • Nenhuma licença encontrada para o servidor. Neste caso, somente uma sessão é permitida.

Avisos

Avisos são atribuídos no intervalo de 6000 a 6999.

6000	A ação não pode ser executada; configuração adicional necessária.
------	---

Informações adicionais

Integração

A abordagem de serviços da Web do ModusOne faz a implementação em um número praticamente ilimitado de cenários de integração possíveis.

A arquitetura de serviços da Web fornece opções altamente flexíveis para integração técnica e corporativa de processos de correspondência em uma empresa.

Integração técnica

A interface padrão dos serviços da Web do ModusOne é a camada da interface SOAP.

Isso proporciona a opção de integração mais profunda possível para serviços da Web. Qualquer linguagem de programação que pode processar SOAP pode ser usada para desenvolver Clientes ModusOne nativos.

Tanto a integração no lado do servidor quanto no lado do cliente pode ser implementada com base nesta interface. A integração em um aplicativo da Web também é possível, por exemplo, pela incorporação direta em Páginas de servidor ativo ou pela integração de chamadas em serviços da Web do ModusOne como links da Internet.

Integração corporativa

A grande vantagem da integração corporativa dos processos de correspondência do serviço da Web do ModusOne é a enorme flexibilidade que ela oferece.

A geração de correspondência com o ModusOne consiste em várias atividades interligadas complementares e colaborativas. Essas atividades podem ser definidas para execução automática ou podem ser iniciadas e manipuladas manualmente por um usuário.

A sequência em que essas atividades são pré-formadas é definida pela sua dependência dos resultados de atividades anteriores.

Os serviços da Web do ModusOne Web permitem acesso granular a essas atividades individuais.

Isso facilita a eliminação de quaisquer etapas desnecessárias ou a repetição de uma ou mais atividades conforme exigidas individualmente. O conceito todo é direcionado à adaptação flexível de procedimentos técnicos para requisitos corporativos específicos.

Considere o seguinte cenário como exemplo. A parametrização da recuperação de dados é muito complexa e implementada sobre várias páginas Web interdependentes. A seleção de documentos, entrada manual de variáveis e edição de documentos não são desejadas. Somente documentos de controle devem ser exibidos e, se corretos, arquivados diretamente em seguida.

Com os serviços da Web do ModusOne, a implementação de um cenário como esse torna-se bem simples.

Repositório do ModusOne

O repositório do ModusOne serve como fundamento básico dos serviços da Web do ModusOne.

Trata-se de onde todas as definições do são armazenadas e de onde são recuperadas. Essas definições podem ser divididas em diferentes categorias:

- Dados relevantes a processos de correspondência, como blocos de texto do pacote e modelos de documentos.
- Dados da área de gerenciamento de saída, como impressoras e formulários.
- Dados relevantes ao sistema

As informações podem ser obtidas dessas áreas usando uma interface especial. Isso permite que um aplicativo de terceiro exercite funções específicas do ModusOne, por exemplo, seleção do sistema ou seleção do pacote incluindo hierarquia de pastas.

Processos de correspondência

Um processo de correspondência consiste em várias atividades interligadas complementares e colaborativas. Essas atividades podem ser definidas para execução automática ou podem ser iniciadas e manipuladas por um usuário ou aplicativo de terceiro.

As atividades também podem ser definidas para execução simultânea.

Como gerenciar um processo de correspondência

Um processo de correspondência é gerenciado em uma tabela de banco de dados e em um sistema de arquivos, sendo identificado por uma chave única. A chave em si pode ser definida por um aplicativo de terceiro.

Normalmente, o ciclo de vida de um processo de correspondência é limitado à criação e impressão de documentos. O termo impressão é entendido aqui para se referir à passagem de documentos para um sistema de gerenciamento de saída, como Odin.

Também é possível salvar um processo de correspondência para reiniciar a manipulação posteriormente. Isso é feito ao atribuir um nível de status interno a um processo.

O status de um processo de correspondência

Um novo processo sempre é criado com o status 1 (em andamento).

Quando um processo é fechado, o status pode ser definido como 0 (salvo, mas aberto para processamento posterior) ou 2 (o processamento está concluído e, em princípio, pode ser excluído).

Ações de um processo de correspondência

Um processo de correspondência é composto das seguintes ações:

- Criar um processo de correspondência
- Especificar parâmetros para recuperação de dados (opcional)
- Recuperação de dados (opcional)
- Aprimorar a estrutura do pacote (pacote dinâmico, opcional)
- Configurar documentos (opcional)
- Definir variáveis manuais (opcional)
- Gerar documentos (opcional)
- Editar documentos (opcional)
- Imprimir e arquivar (opcional)
- Fechar um processo de correspondência

Índice

A	
Ações	4
Ações incorporadas no XML do objeto	24
Arquitetura.....	1
Atributos especiais da ação DATASELECTION	13
Atributos especiais da ação PRINTDOCUMENTS	13
Avisos.....	62
C	
ClearCache	32
Códigos de erro	62
Comandos e seus resultados	4
D	
Doc_GetFile_Mime.....	33
Doc_SetFile_Mime	34
E	
Endereço de um objeto ModusOne em um pacote.....	4
Erro XML	32
Estado da propriedade da ação	12
G	
GetRoles	60
GetUsers	61
GetUsersOfRole	60
I	
Identificador do processo	1
Informações do processo	7
Informações sobre encaminhamento de processo.....	11
Informações sobre XML.....	28
Integração	62
Introdução.....	1
L	
Logoff.....	35
Logon.....	34
Logon XML	29
O	
Obj_GetStructure.....	35
Obj_SetStructure	36
Obj_Toggle.....	37
Obter e definir variáveis de pool.....	14
Opções ao encaminhar um processo	15
Opções ao fechar um processo	15
Opções de processo	2
Opções XML de MwsNavigTree.....	50
Opções XML de Processlist	16
P	
Ping.....	37, 50, 61
Process_Close.....	38
Process_Create	39
Process_Delete.....	40
Process_Forward.....	41
Process_GetInfo	42
Process_GetLastError.....	42
Process_GetList.....	43
Process_Open	44
Process_SetData	44
Process_Start	45
Processos de correspondência.....	63
Propriedade da ação configurada.....	12
R	
Rep_GetBinFile_Mime	51
Rep_GetDataProviderDefinition.....	51
Rep_GetFolderContent.....	52

Rep_GetForms	53	XML de MwsFolderContent.....	49
Rep_GetItemInfo	53	XML de MwsItemInfo.....	46
Rep_GetLookupObjects	54	XML de MwsLookupResult.....	49
Rep_GetLookupValue	55	XML de MwsNavigList.....	47
Rep_GetNavigList	56	XML de MwsNavigTree	47
Rep_GetNavigTree.....	56	XML de MwsSystemList	47
Rep_GetPrinters.....	57	XML de MwsValueHelpDefinition	50
Rep_GetSystem	58	XML de Processlist.....	19
Rep_GetSystemList	58	XML de RoleList.....	59
Rep_GetValueHelpDefinition.....	58	XML de Systemdef	46
Repositório do ModusOne.....	63	XML de UserInfo	29
Resultado da execução com êxito	62	XML de UserList para GetUsers.....	59
V		XML de UserList para GetUsersOfRole.....	60
Valores de retorno de funções do MWS.....	61	XML do objeto	20
X			
XML de MwsDataProviderDefinition	49		