



XML SERVICES

ARTSOFT

Índice

XML Interface	4
<i>Objectivo</i>	4
Modo de funcionamento	4
Caracteres ‘escape’ do XML	4
Parâmetros gerais dos Serviços WEB	4
Retorno dos Serviços WEB	5
Serviços Básicos	7
<i>ArtDB/_WSList</i>	7
<i>ArtDB/_DBTables</i>	7
<i>ArtDB/_tblDesc</i>	8
<i>ArtDB/_fnList</i>	10
<i>ArtDB/Cambios</i>	10
<i>ArtDB/CfgEnums</i>	10
<i>ArtDB/SysTime</i>	11
<i>ArtUsr/TestPsw</i>	11
<i>ArtUsr/Login (Só ArtCMD)</i>	11
<i>ArtUsr/SetPass (Só ArtCMD)</i>	12
<i>Notify/SendMsg</i>	12
Serviços de Query	13
<i>Queries/Query</i>	13
<i>Queries/SQLQuery</i>	15
Serviços relativos a Documentos	17
<i>DocFch/DocInsert</i>	17
<i>DocFch/DocUpdate</i>	18
<i>DocFch/DocSignature</i>	19
<i>DocFch/DocDelete</i>	19
<i>DocFch/DocRemove</i>	20
<i>DocFch/DocForceR</i>	20
<i>DocFch/GetImage</i>	20
<i>DocFch/SetImage</i>	21
<i>DocFch/DocPrices</i>	21
<i>DocFch/DocsByEnt</i>	23
<i>DocFch/DocStatus</i>	23
<i>DocFch/DocPrint</i>	24
<i>DocFch/DocPrintEx</i>	28
<i>DocFch/DocPrintEtiqEx</i>	28
<i>DocFch/CfgDocum</i>	28
Serviços relativos a Artigos	29
<i>StkFch/Update</i>	29
<i>StkFch/Delete</i>	29
<i>StkFch/CfgGroups</i>	30
<i>StkFch/CfgEspec</i>	30
<i>StkFch/CfgMutac</i>	30
Serviços relativos a Terceiros	31
<i>TerFch/Login</i>	31
<i>TerFch/SetPass</i>	31
<i>TerFch/Update</i>	32
<i>TerFch/Delete</i>	32
<i>TerFch/UpdSug</i>	33
Serviços relativos a Eventos de Terceiros	34
<i>CRMHdr/GetTable</i>	34
<i>CRMHdr/GetQualif</i>	34
<i>CRMHdr/Insert</i>	35
<i>CRMHdr/Update</i>	35

<i>CRMHdr/Delete</i>	36
<i>CRMHdr/Status</i>	36
<i>CRMHdr/GrupTask</i>	37
<i>CRMHdr/PersTask</i>	38
<i>CRMHdr/FollowUp</i>	38
<i>CRMHdr/GetImage</i>	38
<i>CRMHdr/SetImage</i>	38
<i>VndFch/Update</i>	39
Serviços relativos a Contabilidade e Contas Correntes	39
<i>CtaLan/CurrAcc</i>	39
<i>CtaLan/OpenDoc</i>	39
<i>CtaLan/DocType</i>	40
<i>CtaLan/DocRecv</i>	40
<i>CtaLan/RecvDoc</i>	40
<i>CtaLan/GetImage</i>	42
<i>CtaLan/SetImage</i>	42
<i>CtaLan/LanDelete</i>	42
<i>CtaLan/LanUpdate</i>	42
<i>CtaFch/UpdateAccounts</i>	43
<i>CtaFch/DeleteAccounts</i>	43
Serviços relativos a Recursos Humanos	45
<i>GrhEmp/Login</i>	45
<i>GrhEmp/SetPass</i>	45
<i>GrhEmp/Update</i>	45
<i>GrhEmp/UpdSug</i>	46
<i>GrhEmp/UpdEvent</i>	47
<i>GrhEmp/DelEvent</i>	47
<i>GrhEmp/UpdEventCad</i>	47
<i>GrhEmp/DelEventCad</i>	47
<i>GrhlInd/Login</i>	48
<i>GrhlInd/SetPass</i>	48
<i>GrhlInd/Update</i>	48
<i>GrhlInd/UpdSug</i>	48
<i>GrhlInd/UpdEvent</i>	48
Apêndice A - XMLReports	49
<i>Funções internas de XMLReports</i>	50
Apêndice B - Lista de variáveis globais ArtSOFT	52
Apêndice C - Lista de funções base ArtSOFT	53
Apêndice D - Lista de Funções	54
Apêndice E - Formulários HTML	56
<i>Variáveis e Fórmulas</i>	57
<i>Formatação de campos</i>	57
<i>Exemplo de formulário HTML</i>	58
Apêndice F - Informação adicional sobre chaves de tabelas ArtSOFT	59
<i>Lançamentos de Contabilidade (CtaLan)</i>	59
Apêndice G - Autenticação no servidor	60
<i>Modo produção</i>	60
<i>Modo debug</i>	61
<i>Teste de comunicação</i>	61
<i>Algoritmos de validação de clientes</i>	61
Apêndice H - Servidor multiempresa	62
Apêndice I - Comunicação Segura	63
Apêndice J - Configuração do servidor	64
<i>Lista de DLL's, serviços e dependências</i>	64
Apêndice K - Validação de Logins	65
Apêndice L - Status WinSock	67
Apêndice M - ArtCmdWS	68
<i>Interface ArtCmdWsDotNet</i>	69
Apêndice N - Controlo de versões	70

XML Interface

Objectivo

Disponibilização de serviços XML específicos, podendo ser chamados directamente no ArtSOFT, via plugins, ou executados a partir de um servidor sobre protocolo HTTP.

Modo de funcionamento

Cada função recebe uma string XML, e retorna o resultado do seu trabalho também numa string XML, e, em alguns casos, numa string HTML.

Caracteres ‘escape’ do XML

A sintaxe do XML utiliza os caracteres '<', '>', '&' '' e '''' (menor, maior, &, apóstrofo e aspa), pelo que não é possível utilizá-los nos comandos. Quando forem necessários, devem ser ‘codificados’ com as sequências '<', '>', '&', ''' e '"', encarregando-se o ‘parser’ de XML de os descodificar. Aos caracteres substitutos costuma chamar-se ‘sequências de escape’¹.

Em termos práticos, e partindo do pressuposto que seria necessário escrever este ‘statement’ SQL :

```
SELECT * from tableX where coluna1 > 10 AND coluna1 < 20 AND nome1 like '%nome'
```

deveria ser escrito desta forma²:

```
SELECT * from tableX where coluna1 > 10 AND coluna1 < 20 AND nome1 like &apos;%nome&apos;;
```

Parâmetros gerais dos Serviços WEB

O primeiro node XML (a root) pode conter um conjunto de atributos que indiquem como se pretende que o serviço seja feito, ou obter a resposta do serviço:

Todos os serviços	
status='[N] S'	Solicita que seja enviada a descrição textual do erro, em complemento ao enumerator já devolvido (rc='xxxxxxxx'), sempre que haja qualquer erro. Serve para ajudar um developper a perceber melhor o que aconteceu, mas poderá não ser suficientemente explícita para mostrar ao utilizador, em determinados contextos.
retid ='[N] S'	Solicita que seja devolvido o ID dos registos processados (autoincremento de artigo, terceiro, documento, nº do evento, etc)
Serviços de inserção de dados	
id='xxx'	Em funções que envolvam terceiros, esta propriedade indica qual o nº de registo destino. Se esta propriedade for enviada e entityId também, irão ser confrontadas para verificar se pertencem à mesma entidade. Se não for o caso, será devolvido o erro ‘ErrInvEntityID’. Esta propriedade pode ser ‘injectada’ por um servidor que tenha validado o terceiro, assegurando-se assim de que não houve adulteração ou injecção de um ‘entityID’ falso. No caso de fichas adicionais (MoreInfo id='xx') , esta propriedade pode também referir o nº de registo de artigo, vendedor, conta, empregado ou independente, que será validada com o campo ‘RegID’ do registo ‘FDUxxx’.
entityID='C F N [R]:xxxx'	Em funções que envolvam terceiros, esta propriedade especifica uma entidade destinatária deste comando. Se forem especificados outros parâmetros relativos a uma entidade (ex: nº de cliente num cabeçalho de factura ou evento) e diferirem deste, o comando será rejeitado por razões de segurança. C:xxx - Cliente x, F:xxx - Fornecedor x, N:xxx - NIF de terceiro, R:xxx - nº de registo (autoincremento) da ficha de terceiro, sem prefixo - nº de registo.

¹ O legado da informática sempre usou sequências de escape para controlo de posicionamento de cursores no ecrã, controlo do tipo de letra em impressoras, etc, em que todas essas sequências começavam com o carácter ‘Escape’ (decimal 27 ou 0x01B), e uma tecla específica para isso, a tecla ‘Esc’.

² No caso dos statement SQL, veremos na respectiva secção que existe uma alternativa mais elegante para esta sintaxe...

contact='xx.yy'	Em funções que envolvam terceiros e realizadas a partir de login de um contacto, esta propriedade indica qual o nº de contacto que efectuou o login. Esta propriedade pode ser ‘injectada/modificada’ por um servidor de validação do contacto do terceiro.
trans='Y T S V 1 N F 0'	Permite especificar que a inserção de dados deve ser feita sob uma transacção. É aplicável quando se pretende que, quando se verifica um erro, o serviço elimine todas as operações feitas anteriormente, mantendo a integridade da base de dados. A indicação da transacção pode ser feita parcialmente, ao nível de uma sub-tag, indicando que só as operações ocorridas nesta devem ser transacionais. Ex. em clientes: <Contacts trans='S'> indicaria que apenas a lista de contactos deve ser transacional, ou seja, insere os contactos todos, ou em caso de erro num deles, não insere nenhum.

Na especificação de uma lista de campos (colunas), esta pode ser feita na forma de nós XML (<tag>valor</tag>), o que permite que em qualquer registo se envie apenas os campos disponíveis, mas permite também a especificação de uma lista de campos, a serem enviados, e depois, estes possam ser enviados serializados, separados por um caracter separador/delimitador. Exemplo:

```
<BaseAddr sep=';' fList='Ter.NrIdFisc,Ter.Nome,Ter.Morada,Ter.Localid,Ter.CPPais'>
<Rec>500862273|Clube de Campismo de Lisboa|Rua Nova nº15|Alcântara|1375-111 LISBOA</Rec>
<Rec>500862284|Clube Caravanistas do Porto|Rua Seis nº33|Ribeira|4101-231 PORTO</Rec>
<Rec>500862145|Clube de Motars do Alentejo|Rua de Évora nº31|Beja|7001-911 BEJA</Rec>
</BaseAddr>
```

O separador ‘sep’ permite indicar qual o caracter usado para a separação de campos, que, por omissão, é assumido ‘;’.

Caso se usem caracteres especiais, estes podem ser especificados como sequencias de escape do tipo ‘C/C++, C#, Java, ex: \a \b \r \n, ou em hexadecimal: \xHH’.

Retorno dos Serviços WEB

Um serviço devolve o resultado do seu trabalho numa string XML, que pode ter várias formas, consoante o tipo de serviço, e o tipo de retorno:

Erro	Parâmetros	Exemplo de retorno	
Sem erros	retID='N' status='?'	<Entity/>	
Registo inexistente	retID='?' status='N'	<Entity rc='RecordNotExist' />	
Registo inexistente	retID='?' status='S'	<Entity rc='RecordNotExist'>Nº de registo não existe</Entity>	
Erro de inserção / update / delete, na base de dados	retID='N' status='N'	<Entity err='3'> ← nº de erros totais <OtherAddr err='1'> ← nº de erros deste nó <Rec id='123' rc='DuplicatedKey' /> </OtherAddr> <Contacts err='2'> ← nº de erros deste nó <Rec id='123' rc='DuplicatedKey' /> <Rec id='456' rc='DuplicatedKey' /> </Contacts> </Entity>	
Erro de inserção / update / delete, na base de dados	retID='N' status='S' trans='S'	<Entity err='1'> <BaseAddr err='1'> <Rec id='1' rc='DuplicatedKey'>chave duplicada</Rec> </BaseAddr> <OtherAddr rc='OperationAbort'>Operação cancelada devido a transacção abortada</OtherAddr> <Contacts rc='OperationAbort'>Operação cancelada devido a transacção abortada</Contacts> <MoreInfo rc='OperationAbort'>Operação cancelada devido a transacção abortada</MoreInfo> </Entity>	
Operação com sucesso	retID='S' status='?'	<Entity> <BaseAddr fList='Ter.NrIdFisc,Ter.Nome'> <Rec>501848487,Clube de Campismo Lisboa</Rec> </BaseAddr> <OtherAddr fList='Ter.NrFilial,Ter.Morada'> <Rec>1,Rua de São Bento</Rec> <Rec>5,Rua de Santa Apolónia</Rec> <Rec>9,Praça de Santo António</Rec> </OtherAddr> <Contacts fList='TpContacto,NrContacto,Nome'> <Rec>1,1,António Almeida</Rec> <Rec>1,2,Joaquim Jeremias</Rec> </Contacts> <MoreInfo fList='NrCpo,Texto'> <Rec>25,Texto 1 para ficha adicional</Rec> <Rec>25,Texto 2 para ficha adicional</Rec> <Rec>25,Texto 3 para ficha adicional</Rec> </MoreInfo> </Entity>	<Entity> <BaseAddr> <Rec id='1234' /> </BaseAddr> <OtherAddr> <Rec id='11' /> <Rec id='12' /> <Rec id='13' /> </OtherAddr> <Contacts> <Rec id='50' /> <Rec id='51' /> </Contacts> <MoreInfo> <Rec id='1234' /> <Rec id='1235' /> <Rec id='1236' /> </MoreInfo> </Entity>

Operação com alguns erros	retID='N' status='S'	<pre> <Entity> <BaseAddr fList='Ter.NrIdFisc,Ter.Nome'> <Rec>501848487,Clube de Campismo Lisboa</Rec> </BaseAddr> <OtherAddr fList='Ter.NrFilial,Ter.Morada'> <Rec>1,Rua de São Bento</Rec> <Rec>5,Rua de Santa Apolónia</Rec> <Rec>9,Praça de Santo António</Rec> </OtherAddr> <Contacts fList='TpContacto,NrContacto,Nome' /> <Rec>1,1,António Almeida</Rec> <Rec>1,2,Joaquim Jeremias</Rec> </Contacts> <MoreInfo fList='NrCpo,Texto'> <Rec>25,Texto 1 para ficha adicional</Rec> <Rec>25,Texto 2 para ficha adicional</Rec> <Rec>25,Texto 3 para ficha adicional</Rec> </MoreInfo> </Entity></pre>	<pre> <Entity err='3'> <OtherAddr err='1'> <Rec nr='2' rc='DBKey_Dupl'> chave duplicada</Rec> </OtherAddr> <Contacts err='1'> <Rec nr='1' rc='DBKey_Dupl'> chave duplicada</Rec> </Contacts> <MoreInfo err='1'> <Rec nr='4' rc='DBKey_Dupl'> chave duplicada </Rec> </MoreInfo> </Entity></pre>
Operação com alguns erros	retID='S' status='N'	<pre> <Entity> <BaseAddr fList='Ter.NrIdFisc,Ter.Nome'> <Rec>501848487,Clube de Campismo Lisboa</Rec> </BaseAddr> <OtherAddr fList='Ter.NrFilial,Ter.Morada'> <Rec>1,Rua de São Bento</Rec> <Rec>5,Rua de Santa Apolónia</Rec> <Rec>9,Praça de Santo António</Rec> </OtherAddr> <Contacts fList='TpContacto,NrContacto,Nome' /> <Rec>1,1,António Almeida</Rec> <Rec>1,2,Joaquim Jeremias</Rec> </Contacts> <MoreInfo fList='NrCpo,Texto'> <Rec>25,Texto 1 para ficha adicional</Rec> <Rec>25,Texto 2 para ficha adicional</Rec> <Rec>25,Texto 3 para ficha adicional</Rec> </MoreInfo> </Entity></pre>	<pre> <Entity err='4'> <BaseAddr err='1'> <Rec rc='DB_KeyNMod' /> </BaseAddr> <OtherAddr err='1'> <Rec ID='1369' /> <Rec rc='DBKey_Dupl' /> <Rec ID='1370' /> </OtherAddr> <Contacts> <Rec ID='22' /> <Rec ID='33' /> </Contacts> <MoreInfo err='2'> <Rec rc='DBKey_Dupl' /> <Rec rc='DBKey_Dupl' /> <Rec ID='266578' /> </MoreInfo> </Entity></pre>

Na informação de retorno alguns serviços poderão ainda incluir o atributo ‘IDEx’ que conterá informação adicional sobre um determinado erro, como, por exemplo, o conteúdo de um determinado parâmetro errado. A documentação do serviço referirá quando isso ocorrer.

Serviços Básicos

É possível inquirir quais os recursos disponíveis no servidor, através de um conjunto de serviços:

ArtDB/ WSList

Informa quais os serviços disponíveis no serviço onde se conectou, ou se for indicado um determinado serviço, qual o protótipo de string XML necessário como input.

Parâmetro	Resultado
<wslist />	<pre> <wsList ttl="44"> <ArtDB> <_DBTables/> <_FnList/> <_TblDesc/> <_WSList/> </ArtDB> <ArtFDU> . . </ArtFDU> <CrmHdr>...</CrmHdr> <DocFch>...</DocFch> <TerFch>...</TerFch> </wsList> </pre> <p>← ttl=nº de serviços disponíveis ← Namespace ArtDB ← serviços do namespace ArtDB ← (neste caso, 'ArtDB/_FnList') ← . . . ← . . . ← fim do namespace ArtDB ← Namespace ArtFDU ← serviços do namespace ArtFDU ← fim do namespace ArtFDU ← Outros namespaces</p>
<wsList service='TerUpdate' />	<pre> <!-- TerUpdate --> <Entity> <BaseAddr> <!-- fields --> </BaseAddr> <OtherAddr> <Rec ID='1'> <!-- fields --> </Rec> <Rec ID='2'> <!-- fields --> </Rec> </OtherAddr> <Contacts> <Rec ID='1.2'> <!-- fields --> </Rec> <Rec ID='1.3'> <!-- fields --> </Rec> </Contacts> <MoreInfo> <Rec> <!-- fields --> </Rec> <Rec> <!-- fields --> </Rec> </MoreInfo> </Entity> </pre> <p>← nome do serviço ← Terceiro ← Morada principal ← Campos da morada principal ← Moradas adicionais ← Morada 1 ← Campos da morada 1 ← Morada 2 ← campos do morada 2 ← Contactos do terceiro ← Contacto 1.2 ← Campos do contacto 1.2 ← Contacto 1.3 ← Campos do contacto 1.3 ← Informação adicional (FDU) ← Informação adicional x ← Campos da informação adicional ← Informação adicional y ← Campos da informação adicional</p>

ArtDB/ DBTables

Informa a lista de tabelas disponíveis no serviço onde se conectou.

Parâmetro	Resultado
<DBTables />	<pre> <DBTables ttl="5"> <TerFch nr="14" key="7" ver="31" log="0">Terceiros</TerFch> <TerCon nr="15" key="2" ver="20" log="0">Terceiros/Contactos</TerCon> <VndFch nr="23" key="3" ver="1" log="0">Vendedores</VndFch> <CrmHdr nr="83" key="9" ver="21" log="0">Eventos de Terceiros</CrmHdr> <CrmDet nr="84" key="1" ver="0" log="0">Eventos de Terceiros-Detalhe</CrmDet> </DBTables> </pre>

O atributo 'nr' informa qual o nº interno da tabela no ArtSOFT, que poderá ser necessário como chave em tabelas que usem esta como 'foreign key'.

O atributo 'key' informa o nº de chaves *lógicas* existentes nessa tabela. *Nota: Poderão existir mais ou menos chaves lógicas que físicas, consoante a tabela.*

O atributo 'ver' indica a versão da tabela, e é meramente informativo (apenas para debug).

O atributo ‘log’ informa se a tabela está marcada como rastreável nas operações feitas sobre ela, na tabela ‘Artlog’.

O texto do nó é o nome comum da tabela.

ArtDB/ *tblDesc*

Informa a lista de chaves e campos da tabela especificada. Para o caso de tabelas de configuração presentes no ficheiro ArtTBE (um ‘tablespace’ que agrupa centenas de tabelas), é necessário especificar num atributo “enum='xxx’” qual a tabela pretendida (de momento apenas está disponível a tabela de configuração de documentos, com o enum ‘tblDoc’, devendo o pedido ser feito assim: <fldList table='ArtTbe' Enum='tblDoc' />).

Parâmetro	Resultado
<tblDesc table='TerCon' [enum='...'] />	<pre> <TerCon> <keys> ← Lista de chaves <AutoInc name="Contactos/AutoIncremento">TerCon AutoInc=1234</AutoInc> <Contact name="Terceiros/Contactos" >TerCon Contact NrTerc=1234 TpCont=1</Contact> <DataAniv name="Data de Aniversário">TerCon DataAniv=20010101:20091231</DataAniv> </keys> <fields ttl="20"> ← Lista de campos (colunas) <NrCli type="UIInt" size="4">Nº de Cliente</NrCli> <NrFor type="UIInt" size="4">Nº de Fornecedor</NrFor> <Filial type="UIInt" size="2">Nº de Filial</Filial> <TpContacto type="UIInt" size="1">Tipo de Contacto</TpContacto> . . </fields> </TerCon> </pre>
<tblDesc table='CtaLan' />	<pre> <CtaLan> <keys> ← Lista de chaves <AutoInc name="Nº de lançamento de contas">CtaLan AutoInc=1:9999</AutoInc> <ExtrDiar name="Lançamentos de um diário">CtaLan ExtrDiar Mes=5 Diario=21</ExtrDiar> . . <Links name="Ligações entre lançamentos">CtaLan Links Link=xxxxxx NrLanc=1234</Links> </keys> <fields ttl="67"> ← Lista de campos (colunas) <Cta.CtaEd type="*Str" size="31">Nº de Conta Editado</Cta.CtaEd> <Data.Lanc type="Date" size="4">Data de Lançamento</Data.Lanc> <Flag.LcPend type="UIInt" size="1">Lançamento pendente</Flag.LcPend> <Flag.NaoSald type="Bool" size="1">Flag de Documento não Saldado</Flag.NaoSald> <Lan.Flags type="Hexa" size="4">Flags Lançamento/Hexadecimal</Lan.Flags> <Val.ValorDb type="*Num" size="8" Dec="2">Valor a Débito</Val.ValorDb> <Val.ValorCr type="*Num" size="8" Dec="2">Valor a Crédito</Val.ValorCr> <C_C.Imposto type="Num" size="8" Dec="2">Valor do Imposto</C_C.Imposto> <C_C.NrVend type="UIInt" size="4" >Nº de Vendedor</C_C.NrVend> <Lan.Obs type="*Str" size="1000">Observações</Lan.Obs> <CDU.01 type="*Str" size="1000" array="31">Campo de utilizador 01</CDU.01> </fields> </CtaLan> </pre>

Na lista de chaves (keys), o nome do nó informa qual o nome interno da chave (a usar nos queries) e o atributo ‘name=’ informa qual o nome comum. O valor do nó contém um exemplo de utilização da chave em queries.

Na lista de campos (fields), o nome do nó informa qual o nome do campo, o atributo ‘type’ informa qual o seu tipo, ‘size’ informa qual o seu tamanho e ‘dec’ se presente, informa o nº de decimais que suporta, e ‘array’, se presente, indica que o campo é um conjunto de campos da mesma natureza, que se repetem ‘x’ vezes, e cujo nome varia entre o valor base indicado e o valor superior (no caso de CDU.01, são 31 campos que variam entre CDU.01 e CDU.31).

Se o tipo de campo for precedido de ‘*’ este campo não existe fisicamente nessa tabela, logo, não disponível em queries SQL. O valor do nó informa o nome comum do campo.

Tipos de campo:

Tipo	Descrição	Tamanho (size='xx')
Str	String alfanumérica	xx=nº máximo de caracteres
Image	String com o caminho de um ficheiro de imagem	xx= nº máximo de caracteres
Sound	String com o caminho de um ficheiro de som	xx=nº máximo de caracteres
Date	Data	xx=4 -> AAAAMMDD, xx=2 -> MMDD
Hour	Hora	xx=4 -> HHMMSS, xx=2 -> HHMM
Int	Inteiro positivo ou negativo	xx=1 -> -128 a +127 xx=2 -> -32678 a +32767 xx=4 -> - 2147483648 a 2147483647
UIInt	Inteiro positivo	xx=1 -> 0 a +255 xx=2 -> 0 a +65535 xx=4 -> 0 a 4294967295
Num	Numérico de vírgula flutuante, ‘dec’ informa a precisão em nº de decimais,	xx=4 -> float xx=8 -> double

Hexa	Campo flags em formato hexadecimal. Ex: 'C1A2F3E4'	xx=1 -> 8 bits, xx=2 -> 16 bits, ...
Bin	Campo flags em formato binário, com o bit menos significativo em primeiro. Ex. para 0x17: '10001110'	xx=1 -> 8 bits, xx=2 -> 16 bits, ...
Bool	Campo flag	xx=1

ArtDB/ fnList

Este serviço fornece uma lista das funções disponibilizadas pelo servidor.

Parâmetro	Resultado
<fnList [param='T F'] a) />	<pre> <fnList ttl="14"> <Contab ttl="6"> <Conta/> ... </Contab> <CtaLan ttl="1">...</CtaLan> ← Namespace e nº de funções deste <CrmHdr ttl="1">...</CrmHdr> ← Namespace e nº de funções deste </fnList> </pre>
<fnList function='Conta' />	<pre> <fnList ttl="1"> <CtaFch> <Conta type="N"> ← Namespace e nº de funções deste ← Nome da função e tipo de retorno <Cta_CCu type="A">51:59/23:26</Cta_CCu> <TpSaldo type=" " def="Saldo">Saldo DebMes CredMes SldMes SldDeb SldCred DebAcm CredAcm</TpSaldo> ← b) <TipoMes type=" " def="MesAct">MesAct MesAnt Reab Jan Fev Mar Abr Mai Jun Jul Ago Set Out Nov Dez Enc Reg</TipoMes> ← c) <TipoAno type=" " def="AnoAct">AnoAct Ano-1 Ano-2 Ano-3 Ano-4 Ano-5</TipoAno> ← c) <TpMoeda type="A" def="EUR" len="3"/> ← d) </Conta> </CtaFch> </fnList> </pre>

a) se indicado param='T', serão incluídas todas as funções disponíveis e respectivos parâmetros.

b), c) e d): Parâmetros de entrada e saída das funções:

Nome	Descrição	Valores possíveis
Type='x'	Tipo de retorno, ou tipo de parâmetro	'N'=Numérico 'A'=Alfanumérico ' '=Enum / lista
Def='x'	Valor por omissão (default), se o parâmetro não for indicado.	
Len='x-y:z'	Tamanho do campo	'3' - três dígitos exactos '3.2' - três numéricos e 2 decimais '1-9' - de um a nove dígitos '1-9.2'-de um a nove numéricos, com 2 decimais

ArtDB/Cambios

Fornece os câmbios de da tabela de países/moedas na data do pedido, ou à data especificada. Se não for especificada uma lista de moedas, serão enviados os câmbios para todas as moedas, excepto para a moeda base (exemplo: se moeda base é 'EUR', nenhuma cotação de 'EUR' será enviada).

Input	Output
<cambios [curr='USD;JPY;ABC;...'] [date='currency_date'] />	<pre> <cambios date='aaaamdd'> <curr ID='USD' buy='1.333' sell='1.222' /> <curr ID='JPY' buy='99.12' sell='99.55' /> <curr ID='ABC' rc='NotFound' /> ← se moeda não existir </cambios> </pre>

ArtDB/CfgEnums

Fornece os dados da tabela de campos tipo lista (enums).

Input	Output
<table type='xx' />	<pre> <table name="xxx"> <rec nr="1" id="01">Internet</rec> <rec nr="2" id="02">Rádio</rec> <rec nr="3" dtBeg="19000101" dtEnd="20060131" id="03">Jornais</rec> <rec nr="4" id="04">Revistas</rec> <rec nr="5" id="15">Outdoors</rec> </table> ... <table err="InvalidTable"/> ou <table err="InvalidType"/> ← se type não existir </pre>

ArtDB/SysTime

Fornece a data e hora locais ou UTC do sistema servidor.

Input	Output
<systime [UTC='T F'] />	<systime>2080-10-12T19:00:00</systime>

ArtUsr/TestPsw

Este serviço testa uma palavra-passe indicando se é forte suficiente para garantir o mínimo de segurança contra ataques de dicionário e conjunções de palavras de dicionário.

Com este serviço as aplicações podem testar uma palavra-passe de chamarem um serviço de mudança de palavra-passe, ficando a saber o mesmo resultado, mas ocupando muito menos recursos. O teste do tamanho da palavra-passe é muito fácil, e cada aplicação pode contar o nº de caracteres e decidir se é muito pequena ou não. No entanto, este serviço exige que tenha pelo menos 4 caracteres de comprimento.

As palavras-passe não devem ter menos de 8 dígitos, não devem ser simétricas (ex.: ABCDABCD), invertidas (ex.: ABCDEDCBA) ou exclusivamente numéricas, exclusivamente maiúsculas, exclusivamente minúsculas, exclusivamente vogais, nem combinações destes tipos (ex.: 123ABC), recomendando-se que contenha mistura de numeros, maiúsculas, minúsculas e símbolos.

Input	Output
<root psw='xxx' />	<root rc='OK PswTooShort Symetric Mirrored OnlyNumbers OnlyUppers OnlyLowers OnlyVowels SequenceOfOnly' /> OK=A palavra passe é considerada forte (para o nº de digitos fornecido) PswTooShort=... tem menos de 4 caracteres; Symetric=... é composta por dois segmentos iguais, ex.: ABCDabcd Mirrored=... é composta por dois segmentos simétricos, ex: ABCDdcba OnlyNumbers=... é composta apenas por números OnlyUppers=... é composta apenas por maiúsculas OnlyLowers=... é composta apenas por minúsculas OnlyVowels=... é composta apenas por vogais SequenceOfOnly=... contém dois segmentos de numeros, maiúsculas, minúsculas ou vogais.

ArtUsr/Login (Só ArtCMD)³

Permite a um serviço usar os registo do ArtSOFT para obter dados e permissões sobre os seus utilizadores, validando as suas credenciais sem que a sua palavra-passe tenha que transitar na rede. Para além disso, informa a plataforma ArtSOFT que doravante, todas as operações efectuadas e que necessitem de registo de quem as efectuou, devem ser registadas com este nº de utilizador. Deve receber um XML formatado da seguinte forma:

Input	Output
<login id='user' seed='0123456789012345' digest='FEDBD236CCC8D897947EC9B7B3AD46667947EC' >	<login id='1' [email='user@url.xxx'] perm='11001101...011'> User Name </login>

- O atributo ‘ID’ deve conter o nome de login do utilizador. Este parâmetro serve para localizar qual o registo onde devem ser encontradas os dados (password e respectivas permissões).
- Os atributos ‘seed’ e ‘digest’ e a descrição do algoritmo de login estão descritos no [Apêndice K - Validação de Logins](#).

Para que seja possível configurar permissões de aplicações externas complementares ao ArtSOFT, é necessário criar um ficheiro de texto de nome ‘plugin.ini’ que deve ser colocado na mesma directória do executável ArtSOFT, com o seguinte formato (para a imagem apresentada):

```
[Acessos]
Nome=Plugins
Acesso1=Zonal
Acesso2= Opção 1
Acesso3= Opção 2
Acesso4=Zona 2
Acesso5=Zona 3
Acesso6=Zona 4
```

³ O ArtCMD ou XMLServices efectuam as operações como ‘de sistema’. Porque o servidor é multithread, efectuar Logins aqui não fazem sentido, porque, se existisse um ‘login’ válido seria substituído imediatamente pelo ‘login’ de outra ligação que entretanto chegasse.

ArtUsr/SetPass (Só ArtCMD)

Este serviço permite modificar a palavra-passe de um utilizador ArtSOFT. Este só pode ser chamado depois de efectuado o ‘ArtUsr/Login’ com sucesso. O input e output deste serviço são iguais aos do serviço ‘ArtUsr/TestPsw’ (pág. 11).

Este serviço exige que a palavra-passe tenha no mínimo 4 caracteres, mesmo que o valor definido na ficha do utilizador no ArtSOFT seja inferior.

Notify/SendMsg

Este serviço permite adicionar uma mensagem ao ficheiro de notificações do ArtSOFT, que, para além de ficar registada na tabela, poderá ser enviada através dos meios de notificação de ArtSOFT instalados na empresa (no ArtSOFT, via HTTP, via email, via SMS, etc.)

Input	Output
<pre><message [FileID='StkFch TerFch' RecID='xxx']> <to>+3519988776;addr@someurl.com;Usrl;Usr2=M;Usr3=MT;#3=M</to> <sub>subject</sub> <msg>message</msg> </message></pre>	<pre><message/> ← mensagem registada <message rc='xx' /> ← não registada, rc='xx'</pre>

Parâmetros:

FileID='StkFch TerFch'	O FileID refere-se ao ficheiro sobre o qual se quer enviar a notificação e ao qual esta vai ficar associada.
RecID='xxx'	O RecID refere-se ao numero do registo (autoincremento) sobre o qual a notificação se refere e ao qual esta vai ficar associada.

Este serviço pode ser dirigido a um conjunto diversificado de destinatários:

- Se for indicado um número (precedido ou não de '+'), será entendido como destinatário um nº de telefone, e esta mensagem deve ser enviada por SMS.
- Se for indicado um endereço no formato xxx@yyy.zz, será entendido como destinatário um endereço de correio electrónico, e esta mensagem deve ser enviada por email.
- Se for indicado um identificador de utilizador, será entendido como destinatário o utilizador ‘x’ do ArtSOFT. Adicionalmente poderá indicar-se também ‘User=M|T|MT’, em que ‘M’ significa ‘notificar adicionalmente por email’ e ‘T’ significa ‘notificar adicionalmente por telefone/SMS. Porém, não sendo indicado qual o email ou telefone, estes deverão constar na ficha do utilizador.
- Se for indicado ‘#x=M|T|MT’, será entendido como destinatário o endereço de email ou de telefone/SMS do primeiro contacto do tipo ‘x’ da ficha do terceiro cujo RecID='yyy' (implica obrigatoriamente que FileID='TerFch').

Serviços de Query

Queries/Query

Este serviço permite obter dados de uma tabela ou conjuntos de tabelas, ou informações disponibilizadas por qualquer função, que estejam acessíveis. Recebe como parâmetro uma string XML, compatível com as regras do XMLReports, descritas no Apêndice A - XMLReports (pág. 49). Das várias funcionalidades que o XMLReports possui, este serviço utiliza as não interactivas⁴, permitindo obter dados obtidos a partir de funções que calculam valores, listas obtidas a partir de uma navegação por um conjunto de registo, ou uma mistura de ambos.

Para verificar a lista das tabelas disponíveis, existe o serviço ‘ArtDB/_DBTables’ (pág. 7). Para verificar a lista de chaves (indexes) e campos (colunas) de determinada tabela, existe o serviço ‘ArtDB/_tblDesc’ (pag. 8), e para verificar quais as funções disponíveis, existe o serviço ‘ArtDB/_fnList’ (pág. 10).

Para obtenção de dados de uma tabela, ou de várias tabelas relacionadas, o XML deverá ter um nó do tipo ‘list’, com um atributo ‘query’. Este atributo deverá conter a indicação da tabela / chave onde pretende obter os dados, e consoante esta, deverá fornecer os limites de cada segmento dessa chave. Poderá conter ainda a posição onde terminou um query anterior para continuação, bem como eventuais filtros.

Um primeiro e simples exemplo de um query:

```
<TBL type='list' name='SKU#1' end='xx' query='StkFch|Principal|Codigo=1:10199'>
<defcol>
  <Codigo form='%StkFch.Cod.Editado' />
  <Nome form='%StkFch.Nome.0' />
</defcol>
</TBL>
```

Neste, podemos observar os seguintes elementos:

1. **type=list:** indica que se trata de uma ‘list’, ou ainda, uma tabela fornecida por um query;
2. **name='xxx'** ou **'xxx#yyy'**, neste caso **SKU#1**. Cada linha dessa list irá chamar-se **<xxx>** ou **'xxx000'** a **'xxx999'**, neste caso, **<SKU1>** a **<SKU9>** (deduzindo-se daqui que a query só pode ter no máximo, 9 elementos).
3. **end='xx':** permite indicar o nº exacto de registo pretendido;
4. **query='...':** Os dados serão obtidos a partir da tabela ‘StkFch’, através da chave ‘Principal’ (que contém a lista dos códigos de artigo), no intervalo [1 a 10109].
5. **DefCol:** Secção obrigatória a seguir a uma ‘list’ que contém a definição de uma linha protótipo.
6. **Codigo:** Primeira coluna da ‘list’. Esta pode ter qualquer nome (desde que seja legal em XML).
7. **Nome:** Segunda coluna da ‘list’. Também pode ter qualquer nome.
8. **Form='StkFch.xxx':** fórmula que permita obter o valor pretendido. A fórmula pode referir variáveis (com prefixo ‘%’) de um espaço de nomes (namespace), neste caso StkFch, expressões aritméticas (ex: %StkFch.Prc.Preco.0 * 1.05), ou funções (com prefixo '\$') de manipulação / formatação de dados (ex: \$Left(%StkFch.Nome.0, 20) ou quaisquer outras disponíveis para obtenção de dados ou valores (ex: \$Conta(%CtaFch.NrConta,Saldo)). Neste caso, só fará sentido colocar uma função que receba um parâmetro vindo de uma variável de uma linha do query, senão, a função será chamada sempre com um parâmetro constante, produzindo obviamente um resultado constante em todas as linhas.

Um resultado possível para o query anterior seria:

<pre><TBL next='5457'> <SKU1> <Codigo>1</Codigo> <Nome>Produtos de grande consumo</Nome> </SKU1> <SKU2> <Codigo>1.0</Codigo> <Nome>Produtos Alimentares, Utilidades</Nome> </SKU2> <SKU3> <Codigo>1.0.10</Codigo> <Nome>Azeites, Óleos, Vinagres, Sal</Nome> </SKU3> <SKU4> <Codigo>1.0.10.01</Codigo> <Nome>Azeites</Nome> </SKU4></pre>	<pre><SKU6> <Codigo>1.0.10.01.00002</Codigo> <Nome>Azeites fio de ouro</Nome> </SKU6> <SKU7> <Codigo>1.0.10.02</Codigo> <Nome>Óleos</Nome> </SKU7> <SKU8> <Codigo>1.0.10.02.00001</Codigo> <Nome>Óleo de soja</Nome> </SKU8> <SKU9> <Codigo>1.0.10.03</Codigo> <Nome>Vinagres</Nome> </SKU9> </TBL></pre>
---	---

⁴ As funcionalidades interactivas permitem ao utilizador interagir manualmente com os dados, o que neste caso não faz sentido, por ser um serviço.

```
<SKU5>
<Codigo>1.0.10.01.00001</Codigo>
<Nome>Azeite dourado</Nome>
</SKU5>
```

Como se pode verificar, o serviço query percorreu os primeiros 9 registos da selecção efectuada, e devolveu o resultado, com uma anotação ‘next=xx’ no nó onde tinha sido efectuado o query. Isto significa que, se for pretendido obter mais registos, deve voltar a repetir-se o query indicando o ponto onde este deve retomar. Para isso, o query deve ter a indicação ‘|#’ seguido do valor devolvido em ‘next’:

```
<TBL type='list' ... query='StkFch|Principal|Codigo=1:10199 | #5457'> ...
```

No ‘query’ permite estabelecer os limites inferior e superior da zona onde se pretendem extrair os dados, que será suficiente na maioria dos casos, mas frequentemente serão devolvidos registos sem interesse, pelo que será necessário eliminá-los. Pode fazer-se isso, acrescentando-lhe um filtro. No exemplo seguinte pretende-se a estrutura de 1º nível da hierarquia do ficheiro de artigos:

```
<TBL type='list' ... query='StkFch|Principal|Codigo=1:10199 | ? $IsEqual(%StkFch.Div.NivFam,1)'> ...
```

Resultado:

<pre><TBL> <SKU1> <Codigo>0</Codigo> <Nome>Produtos de grande consumo</Nome> </SKU1> <SKU2> <Codigo>1</Codigo> <Nome>Vestuário e Calçado</Nome> </SKU2> <SKU3> <Codigo>2</Codigo> <Nome>Artigos de Desporto</Nome> </SKU3> <SKU4> <Codigo>3</Codigo> <Nome>Material Eléctrico</Nome> </SKU4></pre>	<pre><SKU5> <Codigo>4</Codigo> <Nome>Equipamentos Domésticos</Nome> </SKU5><SKU6> <Codigo>5</Codigo> <Nome>Automóveis, Acessórios e Peças</Nome> </SKU6> <SKU7> <Codigo>6</Codigo> <Nome>Informática, e Telecomunicações</Nome> </SKU7> <SKU8> <Codigo>9</Codigo> <Nome>Serviços</Nome> </SKU8> </TBL></pre>
--	--

Ao efectuar um query utiliza-se um valor, ou intervalo de valores para cada segmento de chave, mas no caso especial do primeiro segmento, é possível especificar uma lista de intervalos, ou de valores:

```
<TBL type='list' query='StkFch|Principal|Codigo=1:10199;2:20199;3:30199'> ...
<TBL type='list' query='StkFch|Principal|Codigo=10199;20199;30199;40199;50199;60199;70199;80199'> ...
```

(em falta: como utilizar as flags A e O)

```
<TBL type='list' name='SKU#1' end='xx' query='StkFch|Principal|Codigo=1:10199|Flags=ASNSNXXSSNN...'>
```

Um query pode ser usado para obter apenas informação relevante, como, por exemplo, alguns valores contabilísticos, utilizando algumas funções:

```
<?xml version='1.0' encoding='windows-1252' ?>
<?log file='\\Dir\SubDir\LogContas.csv' type='1' ?>
<ELEM>
<Conta type='num' form='$Conta (611;622:623 / 171:175,Saldo,MesAct,AnoAct)'/>
<CCust type='num' form='$CCusto(171;173:177 / 622:624,Saldo,MesAct,AnoAct)'/>
<Dif_1 type='num' form='$Valor(VendasNac)'/>
<Dif_2 type='num' form='$Valor(VendasNac,MesAct,Ano-1)+$Valor(Existencias) ' />
<Dif_3 type='num' form='$Valor(Disponibilidades)' />
<Dif_4 type='num' form='$Valor(Activo)+$Valor(Activo,12,Ano-1)' />
</ELEM>
```

Neste pedido, o serviço deve devolver o resultado de algumas funções, e que gere um ficheiro de log, onde as funções deverão registar os detalhes onde obtiveram os valores. Nota: Nem todas as funções fornecem logs detalhados.

Queries/SQLQuery

Este serviço permite executar um statement SQL ‘Select’. Obviamente, por razões de segurança, apenas é permitido executar statements **Select**. Sendo o resultado de um query sempre uma tabela rectangular com um número fixo de colunas, e um query poder conter um enorme volume de dados, optou-se por codificar o **recordset** resultante numa string com uma lista sequencial de campos separados por um carácter especial (muito fácil de descodificar), evitando assim a grande redundância do XML na definição dos valores de colunas.

Exemplificando: a partir de “<codigo>xxx</codigo><designacao>Texto da designação</designacao>”, irá ser substituído por “xxx|designação”.

Para isso, é possível definir qual o carácter pretendido para o separador de registos, que, por omissão, é o carácter ‘tab’ (0x09). Para manter a descodificação da string muito simples, criou-se uma restrição ao carácter separador: se este aparecer no resultado do query será substituído por um espaço (dai que, por omissão se use o ‘tab’).

Pode pedir-se também a lista de nomes das colunas do recordset, que por omissão, não serão enviadas (pressupõe-se que quando se especifica a lista de colunas pretendida já se sabe quais são os seus nomes). Porém, existem situações em que esta lista pode ser útil, e assim, pode colocar-se o atributo ‘list=’s’, sendo então devolvidos todos os nomes das colunas.

Um query pode conter muitos milhares de registos, que poderia implicar o consumo excessivo de carga do processador ou largura de banda da rede, se não fosse controlado. Assim, o statement deve conter um delimitador do nº de registos (por ex.: Select TOP 1000), afim de prevenir estes problemas. Se este delimitador não existir, o servidor ‘injecta’ automaticamente “TOP 10000” no query. Se o valor indicado for superior a 10000, será devolvido o erro “**TopXxxTooBig**”.

Se for necessário enviar credenciais para estabelecer a conexão à base de dados, estas podem ser enviadas inserindo dois atributos “UID=’user’ e Pwd=’psw’”.

Na resposta é sempre incluido o atributo ‘nrRecs’ que contém o nº de registos contidos na colecção.

Um exemplo de query simples:

```
<root>
  SELECT DivNrFicha,TerNome,TerMorada,TerLocalid,TerCPPais,TerCAE FROM TerFch
  where TerFch.DivNrFicha < 1000 order by TerCAE
</root>
```

Resposta (‘t’ = ‘tab’ ou carácter 0x09):

```
<root nrRecs="99">
<rec>349|SANTOS & COMPANHIA S.A.|Qta Da Boavista|Golegã|2987-678 GOLEGÃ|1210 </rec>
<rec>135|SIMÕES & ROSA, LDA.|Rua da Azenha 13|Sintra|2710-112 SINTRA|1460 </rec>
<rec>412|COMERCIO, IMPORT & EXPORT, LDA.|Doca do Poço do Bispo|Lisboa|1370-555 LISBOA|3111</rec>
...
</root>
```

Query com indicação do carácter separador, ‘list’, UID, PWD e TOP X:

```
<root sep='|' list='s' UID='Master' PWD='SouOMaior' >
  SELECT TOP 100 DivNrFicha,TerNome,TerMorada,TerLocalid,TerCPPais,TerCAE FROM TerFch
  where TerFch.DivNrFicha < 1000 order by TerCAE
</root>
```

Resposta:

```
<root nrRecs="99" fList="DivNrFicha|TerNome|TerMorada|TerLocalid|TerCPPais|TerCAE">
<rec>349|SANTOS & COMPANHIA S.A.|Qta Da Boavista|Golegã|2987-678 GOLEGÃ|1210 </rec>
<rec>135|SIMÕES & ROSA, LDA.|Rua da Azenha 13|Sintra|2710-112 SINTRA|1460 </rec>
<rec>412|COMERCIO, IMPORT & EXPORT, LDA.|Doca do Poço do Bispo|Lisboa|1370-555 LISBOA|3111</rec>
...
</root>
```

Query utilizando o elemento CData do XML:

O elemento CData é aqui muito útil porque permite introduzir a sintaxe SQL sem qualquer modificação, evitando codificações ou erros de transcrição. No exemplo seguinte podemos verificar que assim podem usar-se directamente os caracteres > e <:

```
<root sep='|'>
<! [CDATA[
  SELECT TOP 100 DivNrFicha,TerNome,TerMorada,TerLocalid,TerCPPais,TerCAE FROM TerFch
  where TerFch.CliNumero>0 AND TerFch.CliNumero<1000 order by TerCAE
</root>
```

```
]]>  
</root>
```

Serviços relativos a Documentos

DocFch/DocInsert

Permite inserir um documento, e eventuais registos associados, como artigos, terceiros e Vendedores.

Este serviço permite a inserção de um simples documento, ou de um conjunto de documentos, artigos, terceiros e vendedores necessários para que a lista de documentos seja processada. Esta última opção existe como conveniência para inserções em batch, mas, rapidamente se concluirá que o XML input poderá ter tamanhos muito grandes, não sendo muito adequado para WEBServices, até porque o seu processamento poderá demorar mais tempo que o ‘timeout’ permitido para cada pedido. Nestes casos, aconselha-se a invocar repetidamente cada um dos serviços que processam cada parte da mensagem.

Para inserção de um documento, a estrutura do XML deverá ser:

Secção do XML	Obrig	Descrição
<document atribx='xx' ... >	S	Início da mensagem e eventuais atributos adicionais que sejam necessários
<customer> <address> ... </address> <delivery> ... </delivery> </customer>	N	Secção opcional para a identificação de um cliente que não conste na tabela de terceiros e não se pretende que não venha a constar. Para isso, é necessário registar os dados da sua morada e da eventual morada de entrega do pedido.
<followUp> ... </followUp>	N	Secção opcional para uma lista de follow-ups a serem associados ao documento
<situation> .. </situation>	N	Secção opcional para uma lista de situações a serem associadas ao documento
<docheader> ... </docheader>	N	Permite enviar dados específicos do cabeçalho de um documento. Se este não for enviado, estes serão deduzidos a partir dos elementos de ‘DocItems’, se estes estiverem presentes. Senão, serão assumidos valores por omissão.
<docitems> ... </docitems> ou <onAcc> ... </onAcc>	S	Permite enviar as várias linhas do documento (docItems), ou efectuar um documento de regularização de outro (onAcc), por ex. facturar encomendas ou guias de remessa.
<paylist> </paylist>	N	Secção opcional para especificar uma lista de meios de pagamento
</document>		

Para inserção de um documento e simultaneamente inserir ou actualizar artigos, terceiros e vendedores, a estrutura do XML deverá ser:

Secção do XML	Obrig	Descrição
<message atribx='xx' ... >		Início da mensagem e eventuais atributos adicionais que sejam necessários
<Entity> ... </Entity>	N	Nesta secção pode incluir-se os dados necessários à criação ou actualização de um terceiro, afim de ser processada antes do processamento do documento. Para especificação desta mensagem, consultar o serviço ‘TerFch/Update’.
<SKU> ... </SKU>	N	Nesta secção pode incluir-se os dados necessários à criação ou actualização de um ou vários artigos, afim de ser processada antes do processamento do documento. Para especificação desta mensagem, consultar o serviço ‘StkFch/Update’
<SalesPer> ... </SalesPer>	N	Nesta secção pode incluir-se os dados necessários à criação ou actualização de um vendedor, afim de ser processada antes do processamento do documento. Para especificação desta mensagem, consultar o serviço ‘VndFch/Update’.
<document>...</document>	S	Secção dedicada à mensagem documento, conforme descrita atrás.
</message>		

Exemplo:

```
<document doc='V901' entityID='C:130' retID='s' status='s' trans='s'>
<docheader>
<Doc.Serie>V901</Doc.Serie>
<Doc.Pedido>12345678</Doc.Pedido>
<Data.Docum>20080405</Data.Docum>
<Ter.NrTerc>130</Ter.NrTerc>
<CPag.Cod>1</CPag.Cod>
<MExp.Cod>1</MExp.Cod>
</docheader>
<docitems>
<rec>
<Cod.Codigo>00100100001</Cod.Codigo>
<Qtd.Real>1</Qtd.Real>
<Val.UnBru>1.55</Val.UnBru>
<Desc.LinS>3.00+1.50</Desc.LinS>
<Div.Obs>Observações</Div.Obs>
</rec>
<rec>
<Cod.Codigo>00100100002</Cod.Codigo>
<Qtd.Real>1</Qtd.Real>
<Val.UnBru>1.55</Val.UnBru>
<Desc.LinS>3.00+1.50</Desc.LinS>
<Div.Obs>Observações</Div.Obs>
</rec>
</docitems>
</document>
```

DocFch/DocUpdate

Permite actualizar um documento de acordo com as regras existentes para alteração de documentos.

Este serviço permite a actualização de um documento. Ao utilizar este serviço é importante realçar que são seguidas as regras de alteração de documentos, assim sendo, num documento assinado, pago ou já regularizado só irá ser possível alterar campos estáticos.

Com este serviço é possível alterar o cabeçalho, inserir novos lançamentos, actualizar lançamentos existentes, ou mesmo eliminar lançamentos, desde que essas operações sejam possíveis para o documento sobre o qual se pretende efectuar essas mesmas operações.

É importante referir, que, no caso de se estar a alterar lançamentos com agregações/espécies e mutações, os mesmos terão que ser enviados na sua totalidade, isto é, caso o lançamento tenha agregações ou espécies/mutações e se queira alterar as mesmas, deverão ser enviados quer os campos que se pretende alterar bem como os que se pretende manter.

Para inserção de um documento, a estrutura do XML deverá ser:

Secção do XML	Obrig	Descrição
<document [trans='s' retid='s' status='s']>	S	Início da mensagem e eventuais atributos adicionais que sejam necessários
<docheader ID='Ai_doc' docID='Txxx/NrDoc'> ... Lista de campos do cabeçalho de documento. </docheader>	S	Permite enviar dados específicos do cabeçalho de um documento. Este campo é obrigatório, pois é necessário enviar a identificação de qual o documento que se pretende alterar. Essa indicação pode ser dada através do Autolncremento do documento, AI='Ai_doc' ou através da identificação do documento por série/número do documento, docID='Txxx/yyy'.
<docitems> ... </docitems>	N	Permite enviar as várias linhas do documento (docItems), inserção de novos lançamentos, alteração de lançamentos existentes ou eliminação de lançamentos.
	N	Secção opcional para uma lista de situações a serem associadas ao documento
<paylist> </paylist>	N	Secção opcional para especificar uma lista de meios de pagamento
<docimage> </docimage>	N	Secção opcional para especificar imagens associadas ao documento
</document>		

Exemplo:

```

<?xml version="1.0" encoding="windows-1252"?>
<document status="s" retid="s" trans="s">

<docheader docID="C020/114" retid="s" status='s'>
  <Doc.NrVended>0</Doc.NrVended>
  <Data.Alert>20111208</Data.Alert>
  <Doc.DataCom>20111208</Doc.DataCom>
  <Doc.MsgAlert>Ola teste 2</Doc.MsgAlert>
  <Doc.Obs>Observações Teste - Provisório</Doc.Obs>
</docheader>

<docitems>

  <rec lineID='1' oper='U'>
    <Qtd.Real>10</Qtd.Real>
    <Div.Obs>Por número do lançamento</Div.Obs>
  </rec>

  <rec lancAI='28475' oper='U'>
    <Qtd.Real>10</Qtd.Real>
    <Div.Obs>Por autoincremento</Div.Obs>
    <CDU.01> 1234 <CDU.01>
  </rec>

  <rec lineID='2' oper='D' />

</docitems>

</document>

```

-> ===== Importante =====
 -> Apenas é permitido enviar dados que possam
 -> ser alterados no cabeçalho.

-> =====
 -> Update do lançamento referente ao lançamento
 -> número 1 do documento em causa. Só podem ser
 -> enviados campos que sejam passíveis de
 -> alteração nos lançamentos.(aplica-se a todos
 -> os nós <rec>)

-> =====
 -> Update ao lançamento, do documento em causa,
 -> identificado com o autoincremento 28475

-> Eliminação do lançamento referente ao
 -> lançamento número 2 do documento em causa

DocFch/DocSignature

Permite assinar um documento, ou uma lista de documentos, que estejam como provisórios e que se pretenda que sejam assinados.

```
<document      docAI='xxxx;yyyy;zzzz' |
  docID='TXXX/NNN;TXXX/NNN;TXXX/NNN' |
  status='s' retid='s' />
```

docAI='xxxx;yyyy;zzzz' é uma lista de autoincrements dos documentos que se pretende assinar.
 docID='...' é uma lista de documentos a assinar, identificados por tipo(TXXX) e número (NNN) de documento.

Input:

```
<document docAI='111;222;333' status='s' retid='s' />
          ... ou ...
<document docID='V003/260182900;V003/260182901;V003/34' status='s' retid='s' />
```

Output:

```

Output 1: Para o caso de se ter utilizado docAI
<document status='s' err='3'>
  <succes ID='111' IDEx='V003/40' />
  <errors rc='RecordNotExist' ID='222'>Nº de registo não existe</errors>
  <errors rc='DocPrepareError' ID='333'>O documento não se encontra em modo de elaboração e como tal não
  pode ser assinado</errors>
</document>

Output 2: Para o caso de se ter utilizado docID
<document status='s' err='2'>
  <succes ID='122' IDEx='V003/41' />
  <errors rc='RecordNotExist' IDEx='V003/260182901'>Nº de registo não existe</errors>
  <errors rc='RecordNotExist' IDEx='V003/34'>Nº de registo não existe</errors>
</document>

```

DocFch/DocDelete

Permite anular um documento. Se a série de documentos estiver configurada para anulação de documentos, o documento será marcado como anulado, removendo os registos de C/C, desintegrando stocks e todas as linhas do documento irão manter-se. Se a série não estiver configurada para anulação de documentos, executará as mesmas operações, mas as linhas do documento serão eliminadas.

```
<document ID='XXXX' | docID='XXXX/YYY'  
MotDel='1' status='S' />
```

MotDel=um elemento da tabela de motivos de anulação
de documentos

DocFch/DocRemove

Permite eliminar um documento. Esta operação faz as mesmas operações que DocFch/DocDelete, eliminando o registo do documento. Esta operação não é possível em séries que têm configurado a anulação de documentos. O comando é idêntico, não sendo necessário o atributo ‘MotDel’, não teria interesse em categorizá-lo, uma vez que o documento vai ser eliminado.

```
<document ID='XXXX' | docID='XXXX/YYY' status='S' />
```

DocFch/DocForceR

Efectua a regularização forçada de documentos pendentes⁵ na totalidade ou apenas de algumas das suas linhas. Para efectuar uma regularização forçada, é necessário justificar qual o motivo, que deverá ser um dos elementos da respectiva tabela.

Para regularizar todo o documento:

```
<root docID='Txxx/yyyy' force='id motivo'/>
```

root	Nome (pode usar-se qualquer um)
docID='xxx/yyyy'	Qual o nº de documento pretendido (alternativa: id='xxx', xxx=autoincremento)
force='id motivo'	Motivo da regularização (um elemento da tabela de motivos de regularização forçada)

Para regularizar apenas algumas linhas:

```
<root docID='Txxx/yyyy'>  
  <rec id='1' force='id motivo' />  
  <rec id='5' force='id motivo' />  
  <rec id='9' force='id motivo' />  
</root>
```

root	Nome (pode usar-se qualquer um)
docID='xxx/yyyy'	Qual o nº de documento pretendido (alternativa: id='xxx', xxx=autoincremento)
id='x'	Nº da linha do documento a regularizar
force='id motivo'	Motivo da regularização (um elemento da tabela de motivos de regularização forçada)

DocFch/GetImage

Fornece a(s) imagem(s) associadas a um documento. Cada página contém um elemento com os atributos: type - informa o tipo de imagem; descr - Texto descrição (se existir) associada ao documento. O texto de contém a imagem formatada em Base64.

Exemplos:

Input:

```
<doc docID='E999/123456' />  
ou  
<doc id='123456' />
```

Doc	Nome do nó (pode usar-se qualquer nome)
docID='xxx/yyyy'	Qual o nº de documento pretendido (alternativa: id='xxx')
id='xxxx'	ID do documento, ou autoincremento (alternativa: docID='xx/yy')

Output:

```
<Doc>  
  <img type="jpg" descr="blabla">/9j/4AAQSkZJRgABAQEAcAB4AAD/2wBDAgGBgcGBQgHBwcJCQgHBwcJCQ  
  ...  
  W9ZY0VtzKyEADP8AtFT+FFFFbQpxprljsdFO1CkuWCsjW9ZY0VtzKyEADP8AtFT+FFFFbW9ZY0VtzKyEADP8A/9k=  
  </img>  
  <img type="png" descr="blá-blá">iVBORw0KGgoAAAANSUhEUgAAKUAAACoCAMAAAC2a942H2j5es0cvH2j5
```

⁵ Propostas, Encomendas, etc.

```
...
utJ50nyEnCLKT1n+H2j5es0cvUclAAAAAE1FTkSuQmCCH2j5es0cvH2j5es0cvH2j5es0cvH2j5es0cvH2j5es0c=
</img>
</Doc>
Ou
<Doc err="ErrorID"/>
Erros:
DocImgInvalDocID, DocImgDocNoImage
```

DocFch/SetImage

Permite associar imagem(s) a um documento. Cada página deverá conter um elemento com os atributos: type - informa o tipo de imagem; descr - Texto descrição (se existir) associada ao documento. O texto de contém a imagem formatada em Base64.

Doc	Nome do nó (pode usar-se qualquer nome)
nrDoc='xxx/yyyy'	Qual o nº de documento pretendido (alternativa: id='xxx')
id='xxxx'	ID do documento, ou autoincremento (alternativa: nrDoc='xx/yy')
retid='s n'	Pretende-se que seja devolvido o autoincremento dos artigos processados
status='s'	Solicita que seja enviada a descrição textual do erro

Exemplos:

Input:

```
<doc nrDoc='E999/999999' retid='s' stat='s'>
  <img type="jpg" descr="Imagen 'smile'">/9j/4AAQSkZJRgABAQEAcB4AAD/2wBDAgGBgcGBQgHBw
    ...
    WyXzf166nFWrc+i2LVFFe6c4UUUAUNV/49TWdo/wDrfxoooA6CiiigAooooA//2Q==
  </img>
  <img type="png" descr="Caricatura">iVBORw0KGgoAAAANSUhEUgAAAEcAAABGCAIAAAARjh7tAACACXB
    ...
    fpCrJ7YAAAASUVORK5CYII=
  </img>
</doc>
```

Output:

```
<doc>
  <img id="3"/>
  <img id="4"/>
</doc>
Ou, caso haja erros:
<tag err="MissingImageData"/>
Ou
<tag err="2">
  <img err="InvalidImageType"/>
  <img err="MissingImageData"/>
</tag>
Erros:
DocImgInvalDocID, InvalidImageType, MissingImageData, ErrorSavingImage
```

DocFch/DocPrices

Fornece uma lista de preços para os artigos pedidos, considerando um cliente e uma série de documentos.

Input:

pricelist	Nome do nó (não é obrigatório que tenha este nome, mas é recomendado)
doc='xxx'	Especifica qual a série à qual os preços se referem, ex: S999, V1, V999
dsc='[N] S all'	N: Preço líquido, S: Preço bruto e desconto composto, all=Preço Bruto, e lista de descontos separados por ‘+’
vattx='[N] S'	Indicar qual a taxa de IVA do artigo
vatinc='[N] S'	O preço deve incluir o valor do IVA
stk='[N] S'	Deve ser indicado o stock existente
obs='[N] S'	Deve ser incluído o campo ‘observações’ da ficha de negociações

entityID='C F N [R]:nn'	Identificação do cliente ao qual se pretendem os preços. Se não indicada, os preços fornecidos são os de cliente indiferenciado.
sku='xxxxxx'	Código Principal de Artigo
sku2='xxxxxx'	Código Opcional de Artigo
qty='nnn'	Quantidade à qual se pretende cotação

Exemplo:

```
<pricelist entityID='N:501848487' doc='V1' vatinc='n' vattx='s'
           dsc='s' obs='s' stk='s' retid='s' status='s'>
<item sku='1019910001' qty='4' />
<item sku='S500110002' />
<item sku='S501110006' qty='8' />
<item sku='S502110009' />
</pricelist>
```

Tag / Atributo	Obrig	Descrição
<pricelist		
entityID='N:501848487'	N	Identificação do cliente (por omissão: indiferenciado)
doc='V1'	S	Identificação da série de documentos
vatinc='n' vattx='s'	N	Pretende-se preços sem IVA, e a respectiva taxa
dsc='s'	N	Pretende-se obter preços brutos e desconto composto
obs='s'	N	Pretende-se as observações da ficha de negociações.
stk='s'	N	Pretende-se a existência em stock no armazém da série V1
retid='s'	N	Pretende-se que seja devolvido o autoincremento dos artigos processados
status='s'	S	Solicita que seja enviada a descrição textual do erro
<item sku='1019910001'	S	Indicação do código principal de artigo (para o opcional usar sku2)
qty='4' />	N	Indicação da quantidade para a qual se pretende cotação
<item sku='S500110002' />	N	Indicação opcional de uma lista de artigos e respectiva
<item sku='S501110006' qty='8' />		quantidade
<item sku='S502110009' />		
</pricelist>		

Output:

Item	Nome de cada elemento da lista
id='nn'	Identidade/Autoincremento do artigo
price='nnnnnn.nn'	Preço do item
disc='nn.nn xx.xx+yy.yy'	Desconto ou lista de descontos (consoante dsc='S' ou dsc='all')
vat='nn'	Taxa de IVA
stock='nn'	Saldo real em stock no armazém correspondente à série de documentos doc='xxx'. Só se foi pedido, com stk='S'. Se houver stock negativo, devolve '0'.
avail='nn'	Tal como o anterior, mas devolvendo o saldo disponível.
bonus='nnn'	Quantidade bónus (resultante de uma promoção ou negociação)
skuT='xxxxx'	Referência ou código de artigo do terceiro / cliente
qty='4'	Existe uma negociação com quantidade mínima e o pedido feito com a quantidade 'qty' que é inferior á negociada. Este nó é marcado com rc='WarnOnlyMinQty'.
mingty='nnn'	Quantidade mínima para o preço indicado
timetd='nnn'	Tempo, em dias para entrega (time to delivery)
(valor do nó)	Observações existentes na ficha de negociações

Exemplo:

```
<pricelist>
<item id="1566" rc="WarnOnlyMinQty" price="40.8" disc="0" vat="20" stock="1" avail="1"
      skuC="C8000" qty="4" mingty="5" timetd="10">Observações da ficha de negociação</item>
<item id="1837" price="1000" disc="0" vat="20.5" stock="10" avail="8"
      skuC="CS8000" mingty="2"/>
<item id="1838" price="1000" disc="0" vat="20.33" stock="15" avail="10"
      skuC="P8000"/>
<item id="1836" rc="WarnOnlyMinQty" price="0.55" disc="3.31" vat="20" stock="50" avail="0"
      skuC="X8000" qty="0" mingty="5"/>
</pricelist>
```

DocFch/DocsByEnt

Fornece uma lista de documentos de um terceiro de determinada série, completa, ou apenas com documentos pendentes de regularização. A função ‘DocStatus’ faz o mesmo que esta, mas é especializada apenas num documento, pelo que nesta, apenas serão descritos os parâmetros que aquela não suporta.

Input:

```
<report doc="Txxx" entityID='xxx' begDate='xxx' endDate='xxx' ... ... />
```

doc='Txxx' />	Tipo e série de documentos. Este parâmetro é obrigatório.
entityID='xxx'	Nº de terceiro, cliente, fornecedor, NIF, etc. Este parâmetro é obrigatório.
div='S [N] '	Obter os documentos do terceiro como indiferenciado. Mesmo que exista um registo de cliente / fornecedor, podem ser elaborados documentos para este como terceiro indiferenciado.
begDate	Opcional, e indica a data inicial em que o relatório deve começar.
endDate	Opcional, e indica a data inicial em que o relatório deve terminar.

Outputs:

Input	Output
<report doc='C1' entityID='xxx' begDate='xxx' endDate='xxx' />	<report> <document docID="C001/12519"> <docitems pend="1"/> </document> <document docID="C001/13065"> <docitems pend="1"/> </document> <document docID="C001/13299"> <docitems pend="0"/> </document> </report>
<report doc='C1' entityID='xxx' fwup='s' />	<report> <document docID="C001/12519"> <fwup id="0"/> <docitems pend="1"/> </document> <document docID="C001/13065"> <fwup id="3"/> <docitems pend="1"/> </document> <document docID="C001/13299"> <fwup id="0"/> <docitems pend="0"/> </document> </report>
<report doc='C1' entityID='xxx' fwup='s' detail='s' retid='s' status='s' />	<report> <document docID="C001/12519"> <fwup id="0"/> <docitems pend="1"> <rec id="61" sku="ZAS0401005" ord="4" pend="2"/> </docitems> </document> <document docID="C001/13065"> <fwup id="3">Aprovação Encomenda</fwup> <docitems pend="1"> <rec id="71" sku="S503110000" ord="1" pend="1"/> </docitems> </document> </report>

Os detalhes da função ‘DocStatus’ complementam a descrição desta função.

DocFch/DocStatus

Fornece informação sobre o estado de um documento, de forma abreviada, ou detalhada por cada item, para todos os itens ou apenas para os que estão pendentes de regularização. Esta função faz o mesmo que a anterior, mas é especializada apenas para um documento, e pode ter utilidade quando foi pedida a anterior em modo reduzido, e se pretende mais detalhes sobre um documento em particular.

Input:

```
<root docID="Txxx/yyyy" id='xxx' entityID='xxx' div='s' all='s' fwup='n' level='n' docitems='S' sku='1' />
```

id='xxx'	Autoincremento do documento
docID='Txxx/yyyy'	Tipo, série e número de documento. Se enviados doc e DocID, serão validados
entityID='xxx'	Nº de terceiro, cliente, fornecedor, NIF, etc. Se enviado entityID, será validado se o documento pertence a esta entidade.
div='S [N] '	Documentos de 'entityID' como terceiro diverso/indiferenciado
all='S [N] '	Todos os documentos ou só pendentes
fwup='S [N] '	Pretende-se que seja enviado o follow-up do documento
level='S [N] '	Pretende-se que seja enviado o estado actual do documento
docItems='S [N] '	Pretende-se que seja enviado o estado de regularização de cada linha do documento.
sku='D OIT'	Pretende-se que seja enviado o código principal (default), alternativo (opcional) ou do terceiro.
retID / status	(comportamento standard)

Outputs:

Input	Output
<root docID='Txxx/yyy' />	<root> <docitems pend="1"/> </root>
<document docID='Txxx/yyy' fwup='s' level='s' />	<document> <fwup id="2"/> <level id="4"/> <docitems pend="1"/> </document>
<document docID='Txxx/yyy' fwup='s' level='s' detail='s' />	<document> <fwup id="2"/> <level id="4"/> <docitems pend="1"> <rec sku="S501110004" ord="1" pend="1" /> </docitems> </document>
<document docID='Txxx/yyy' detail='s' sku='2' />	<document> <docitems pend="1"> <rec sku2="5603594000001" ord="1" pend="1" /> ← código opcional de artigo </docitems> </document>
<document docID='Txxx/yyy' detail='s' all='s' />	<document> <docitems pend="1"> <rec sku="S501110004" ord="1" pend="1"/> <rec sku="S501110033" ord="5" pend="0"/> <rec sku="S502110003" ord="2" pend="*1" reason="3"/> ← regularização forçada </docitems> </document>
<document docID='Txxx/yyy' detail='s' all='s' retid='s' />	<document docID="C001/13457" entityID="C:8000"> <docitems pend="1"> <rec id="29093" sku="S501110004" ord="1" pend="1"/> <rec id="29094" sku="S502110003" ord="2" pend="*2" reason="3"/> </docitems> </document>
<document docID='Txxx/yyy' fwup='s' level='s' detail='s' all='s' retid='s' status='s' />	<document docID="C001/13457" entityID="C:8000"> <fwup id="2">Aprovação Financeira</fwup> <level id="4" perc="75">Em negociação avançada</level> <docitems pend="1"> <rec id="29093" sku="S501110004" ord="1" pend="1"/> <rec id="29094" sku="S502110003" ord="2" pend="*2" reason="3">Decisão Suspensa</rec> </docitems> </document>

É de realçar que quando “ all='s' ” todos os itens do documento são apresentados, mesmo aqueles que já não estão pendentes. No entanto, existem dois motivos para que um item não esteja pendente: foi processado (encomendado, facturado, etc.) ou cancelado. Se for necessária essa informação, com “ all='s' ” num item cancelado (com regularização forçada), irá ser apresentada a quantidade processada antecedida de “ * ” e adicionado o atributo ‘reason’ que indica qual a causa da regularização. Se “ status='s' ”, será apresentada também a informação textual do porquê.

DocFch/DocPrint

‘Imprime’ um documento em formato .HTML, mediante transformação das variáveis existentes num formulário indicado no comando, ou parametrizado na série de documentos. Retorna um ficheiro HTML com o resultado, ou uma string XML com a descrição do erro.

Para impressão de um documento através do gerador de relatórios, ver a descrição do serviço DocFch/DocPrint (pág. 28).

Input:

```
<docum doc='23693' docID='V1/12666' id='9130' entityID='C:14997' form='fileName.html' status='s' />
```

status='s'	Em caso de erros, pretende-se a descrição completa do mesmo
doc='xxx'	Autoincremento do documento
docID='Txxx/yyyyy'	Tipo, série e número de documento. Se enviados doc e DocID, serão validados
entityID='xxx'	Nº de terceiro, cliente, fornecedor, NIF, etc. Se enviado entityID, será validado se o documento pertence a esta entidade.
form='htm/filename.html'	Por omissão, é usado o primeiro nome parametrizado na série de documentos, alterado-se a extensão para .HTM, e que deve residir na directória de formulários. Este atributo ignora o que está parametrizado e permite especificar o nome do formulário a usar, e eventualmente um subdirectorio do directorio de formulários.

Output:

O resultado desta função é uma string HTML, com o formato desejado, bastando para isso efectuar o respectivo desenho em .HTML. Como se pode ver, as figuras seguintes representam uma factura e dois talões com designs diferentes, resultantes da transformação desse formulário.

**Empresa de Formação de Utilizador
Artsoft, S.A.**

Avenida da Informática, 333-13º
Portugal Soft Park
1234-567 LISBOA

Telefones: 214326690/98 - Fax: 214326699
web: www.artsoft.pt - email: geral@artsoft.pt
NIF: 501234560, Cap.Social: 100.000€
Inscrição C.R.C. Lisboa,sob o nº 8671

Factura nº 901,000,010
**A Batalhense - Materiais de Construção,
Lda**

Rua D. Nuno Álvares Pereira, nº 4
Alcanadas
2380-003 ALCANENA

Data	Nº Cliente	V/Nº Contrib.	Encomenda	Nº Vend.	Cond. Pagamento			Data Vencim.
Sábado, 9 de Fevereiro de 2008	116	500878250	B0035A	5	Pronto Pagamento			09.02.2008

Código de artigo	Descrição	Qtd.	Preço Unit.	IVA	% Desc.	Total Liq.
31200100001	Azulejo Tejo Ref. 001 15X15	30.0	5.00	20	10	128.99
31200100006	Azulejo Guadiana Ref. 006 15X15	60.0	4.82	20	10	248.68
31200100031	Azulejo Douro Ref. 031 15X15	43.0	4.95	20	10	183.03

IVA %	Incidência	Valor IVA	TOTAIS
0	0.00		Total Iliquido 652.05
5	0.00	0.00	Desc. Produto 586.85
12	0.00	0.00	Desc. Financeiro 560.70
20	560.70	112.14	Total IVA 112.14
Totais	560.70	112.14	TOTAL DO DOCUMENTO *****672.84

Extenso: seiscentos e setenta e dois euros e oitenta e quatro centimos

Doc. 1

**Empresa de Formação de Utilizador
Artsoft, S.A.**
Avenida da Informática, 333-13º
Portugal Soft Park
1234-567 LISBOA

Factura Nº*901,000,010
A Batalhense - Materiais de Construção, Lda
Rua D. Nuno Álvares Pereira, nº 4
Alcanadas
2380-003 ALCANENA

Data	Cliente	V/NIF	Encom.
09.02.2008	116	500878250	B0035A

Vendedor	Cond. Pagamento	Vencimento
5	Pronto Pagamento	09.02.2008

Código	Quant.	IVA	Tot. Iliq.
31200100001	30.0	20	128.99
31200100006	60.0	20	248.68
31200100031	43.0	20	183.03

IVA %	Incidência	Valor IVA
5	0.00	0.00
12	0.00	0.00
20	560.70	112.14
TOTais	560.70	112.14

Total Bruto	652.05
Total Líquido	560.70
Total do IVA	112.14
TOTAL DO DOCUMENTO	672.84

Doc. 2

**Empresa de Formação de Utilizador
Artsoft, S.A.**
Avenida da Informática, 333-13º
Portugal Soft Park
1234-567 LISBOA

Telefones: 214326690/98 - Fax: 214326699
web: www.artsoft.pt - email: geral@artsoft.pt

Factura Nº90100017**
AmoraHiTech - Altas Tecnologias da Amora, Lda
Rua Cidade da Praia, 12
Cruz de Pau
2845-001 AMORA

Data	Cliente	NIF	Pedido
17.10.2008	129	500906360	

Vendedor	Cond. Pagamento	Data Vencim.
0	Pronto Pagamento	17.10.2008

Código	Quant.	IVA	Total Liq.
6017100001	4.00	20	40,000.00

Software de Gestão Artsoft Pack Total
6017100001 4.00 20 40,000.00

Software de Gestão Artsoft Pack Total
6017100001 3.00 20 28,050.00

% IVA	Incidência	Valor IVA
5	0.00	0.00
12	0.00	0.00
21	68,050.00	14,290.50
Totais :	68,050.00	14,290.50

TOTAIS
Total Iliquido: 68,050.00
Total Líquido: 68,050.00
Total de IVA: 14,290.50

Total do Documento
82,340.50

São:oitenta e dois mil, trezentos e quarenta euros e cinquenta centimos

Doc. 3

O *Apêndice E - Formulários HTML* descreve como formatar ficheiros .HTML de forma a serem transformados pelos serviços do ArtSOFT.

DocFch/DocPrintEx

Este serviço imprime um documento através do gerador de relatórios. Para que este serviço esteja disponível, é necessário carregar o ‘DocPrintCmd.DLL’ ou ‘DocPrintSvc.DLL’ (no ficheiro de configuração, na linha ‘ListaDLL’ (ver *Apêndice J - Configuração do servidor*, na pág. 64).

Este serviço (bem como o *DocFch/DocPrintEtiqEx*), poderá ter muita utilidade se for usado através do ArtCMD (versão DocPrintCmd), replicando as mesmas funcionalidades apresentadas pela impressão de documentos do ArtSOFT (selecção de opções de impressão, pré-visualização do documento, etc.). No entanto esta versão não deve ser utilizada através do servidor XML, pois este, normalmente estará a ser executado num servidor de aplicações, não podendo interagir com o utilizador, nem podendo aceder facilmente às impressoras que este disponha no seu posto de trabalho ou remotamente.

Se for necessária a impressão de documentos através do servidor XML, deve usar-se a versão DocPrintSvc.DLL. No entanto, este serviço requer alguma atenção extra...

1. A configuração de impressão de da(s) série(s) a imprimir deve ser feita através do ArtSOFT, e no na configuração do projecto não pode ser usada a opção ‘visualizar’;
2. O serviço só terá disponíveis as impressoras instaladas no servidor.
3. Deverão ser tidas em conta as restrições de licenciamento do List & Label para estes casos.

DocFch/DocPrintEtiqEx

Este serviço imprime etiquetas referentes a lançamentos de um documento através do gerador de relatórios. Tal como para o serviço *DocFch/DocPrintEx*, é necessário carregar os mesmos DLL, de acordo com as instruções já referidas para esse serviço.

Exemplo de utilização:

<pre><docum id='ai_doc' docID='Txxx/yyyy-zzz' [form='etiqueta.jsx'] /></pre>	form='xxx' - Imprimir com o formulário especificado, que por omissão é o parametrizado na série de documentos
--	---

DocFch/CfgDocum

Fornece dados da tabela de configuração de documentos. Os nomes a incluir no atributo ‘Fields’ podem ser obtidos a partir de *ArtDB/_tblDesc* (pág.8), com a tabela ‘ArtTbe’ e enum ‘tblDoc’.

Input	Output
<pre><table type='V1:V999' fields='xx,yy,zz' name='T F' /></pre>	<pre><table> <rec id="E901" arm="11" flags="111100"> Guias Remessa de Fornecedor </rec> <rec id="E903" arm="11" flags="1111000011110000111100000001111"> Facturas de Fornecedor </rec> <rec id="E904" arm="11">Vendas Dinheiro Fornecedor</rec> <rec id="E909" arm="11">Devolução a Fornecedor</rec> </table> ou <table> <rec id="E901" arm="11" nome="G.R.Fornecedor">G.R.Fornecedor</rec> <rec id="E902" arm="11" nome="Fact. Fornecedor">Facturas Fornecedor</rec> <rec id="E909" arm="11" nome="Devol.Fornecedor">Devol. Fornecedor</rec> </table> ou <table err="InvalidParam"/> ou <table err="InvalidGroup"/> ← se group não for 1,2 ou 3</pre>

Serviços relativos a Artigos

StkFch/Update

Permite inserir um artigo, e eventuais registos associados, como redefinições de armazéns, negociações com terceiros, unidades logísticas, equivalências, composições e registos de fichas adicionais.

Para inserção de um artigo, a estrutura do XML deverá ser:

Secção do XML	Obrig	Descrição
<sku rec='xx' ... >	S	Início da mensagem e eventuais atributos adicionais que sejam necessários
<MainRec> <Cod.Codigo>...</Cod.Codigo> <outros campos/> </ MainRec >	N	Secção opcional para a identificação de um artigo que não conste na tabela de artigos. Se esta secção não existir, o atributo rec='ReclDxxx' tem que estar presente no nó SKU.
<Warehouses> <rec id='nrArm'> (redefined sku fields) </rec> <rec nr='nrArm'> (redefined sku fields) </rec> </Warehouses>	N	Secção opcional para a lista de campos do artigo redefinidos no armazém. Se não estiver presente no nó 'rec' o atributo id='nrArm', o campo 'Logis.NrArm' tem que ser válido. Se estiverem os dois presentes, serão validados quanto à sua consistência. Os campos a colocar no espaço fields deverão ser os que tiverem disponíveis na tabela StkArm, que pode ser consultada através do serviço ArtDB/_TblDesc.
<EntityNeg> <Client id='xx.y'> (fields ...) </Client> <Supplier id='xx.y'> (fields ...) </Supplier> </EntityNeg>	N	Secção opcional para uma lista de negociações com clientes ou fornecedores. Se no nó 'Client' ou 'Supplier' o atributo id='xxx.y' não estiver presente, os campos 'Ter.Cli' ou 'Ter.Forn' têm que ser válidos. Se ambos estiverem presentes, serão validados quanto à sua consistência. Os campos a colocar no espaço fields deverão ser os que tiverem disponíveis na tabela StkRft, que pode ser consultada através do serviço ArtDB/_TblDesc.
<Logistics> <rec id='xx'> (fields ...) </rec> </Logistics>	N	Secção opcional para uma lista de Unidades logísticas de artigo. No campo id='xx' deverá ser passado o numero de identificação da unidade logistica que se está a criar (1,2,8, etc). Os campos a colocar no espaço fields, os que fizerem sentido, poderão ser consultados na tabela StkUnl através do serviço ArtDB/_TblDesc.
<Equival> <Rec id='xx'> (fields ...) </Rec> <Rec id='xx'> (fields ...) </Rec> </Equival>	N	Secção opcional para uma lista de artigos equivalentes associadas ao registo de artigo. No campo id='xx' deverá ser passado o numero de identificação do artigo equivalente que se está a criar ou a alterar. De notar que a numeração da identificação do artigo equivalente é inicia a 0. Os campos a colocar no espaço fields deverão ser os que tiverem disponíveis na tabela StkEqu, que pode ser consultada através do serviço ArtDB/_TblDesc.
<Composition> <Rec>(fields ...)</Rec> <Rec>(fields ...)</Rec> </Composition>	N	Secção opcional para uma lista de artigos de produção associados aos registo de artigo. Os campos a colocar no espaço fields deverão ser os que tiverem disponíveis na tabela CmpFch, que pode ser consultada através do serviço ArtDB/_TblDesc. ***** Em Construção *****
<MoreInfo> <Rec>(fields) <Rec>(fields) </MoreInfo> </document>	N	Secção opcional para uma lista de fichas adicionais que se queira associar ao registo de artigo. Os Campos que devem constar no nó Rec podem ser consultados através do serviço ArtFDU/xxxxxx.

StkFch/Delete

Permite eliminar um registo de artigo, caso este registo não tenha lançamentos em documentos pendentes.

Para eliminação de um artigo a estrutura do XML deve ser:

Secção do XML	Obrig	Descrição
<root id='xxx' trans='Y' status='y' retID='y' />	S	Id='xxx' - identificação do registo de artigo (autoincremento) que se pretende eliminar.

StkFch/CfgGroups

Fornece os dados da tabela de grupos de artigos.

Input	Output
<table group='1 2 3' />	<table> <rec fam="101">Equipamentos</rec> <rec fam="102">Sistemas</rec> <rec fam="201">Software pessoal</rec> <rec fam="202">Software professional</rec> <rec fam="203">Software Premium</rec> <rec fam="204">Software Upgrades</rec> </table> ... <table err="InvalidParam"/> ou <table err="InvalidGroup"/> ↵ se group não for 1,2 ou 3

StkFch/CfgEspec

Fornece os dados da tabela de espécies passada como parâmetro no campo type="xxx".

Input	Output
<table type='xxx' />	<table title="Calçado Bebé"> <rec id="1">Azul</rec> <rec id="2">Cor Rosa</rec> <rec id="3">Verde</rec> <rec id="4">Laranja</rec> <rec id="5">Amarelo</rec> </table>

StkFch/CfgMutac

Fornece os dados da tabela de mutações passada como parâmetro no campo type="xxx".

Input	Output
<table type='xxx' />	<table title="Calçado Bebé"> <rec id="0">15</rec> <rec id="1">16</rec> <rec id="2">17</rec> <rec id="3">18</rec> <rec id="4">19</rec> <rec id="5">20</rec> </table>

Serviços relativos a Terceiros

TerFch/Login

Permite a um serviço web usar o registo de contactos do ArtSOFT para controlar contas e acessos de terceiros a partir da tabela de contactos, e validar o registo/password sem que esta tenha que transitar na rede. Deve receber um XML formatado da seguinte forma:

```
<Login id='LBW0XGXL' seed='0123456789012345' digest='FEDBD236CCC8D897947EC9B7B3AD46667947EC' terAcc='x' />
```

- O atributo ‘ID’ deve conter o identificador do registo de contacto do terceiro, fornecido pelo ArtSOFT na respectiva ficha, a partir do botão ‘Acessos’ (ver imagem ao lado). Este parâmetro serve para localizar qual o registo onde devem ser encontradas as credenciais (password e respectivas permissões). Este parâmetro é calculado pelo ArtSOFT, de forma a ter sempre 8 dígitos entre ‘0..9’ e ‘A..Z’ (não diferenciando entre maiúsculas e minúsculas), e ser ‘pseudo-aleatório’, de forma a não poder ser facilmente dedutível, numa empresa cliente, qual o ID de um utilizador a partir de outro, afim de aumentar a segurança do sistema.
- Os atributos ‘seed’ e ‘digest’ e a descrição do algoritmo de login estão descritos no [Apêndice K - Validação de Logins](#).
- O atributo ‘terAcc’ (opcional) informa que se pretende a lista das contas do terceiro que este contacto tem acesso (como cliente, fornecedor ou ambas).



(parte da) janela que contém o ID do contacto.

Output:

```
<Login rc="0" id="1" entityID="1" contact="1.1" flgCC="1000" flgEnc="10000" flgWeb="1111111111111111">
<Cli nr="21.1.1.1.00100">Clientes C/C, Gerais - MN</Cli>
<Cli nr="21.2.1.1.00100">Clientes títulos a receber</Cli>
<Cli nr="21.1.1.2.00100">Clientes C/C, Gerais - UE </Cli>
<Cli nr="21.1.1.3.00100">Clientes C/C, Gerais - OM</Cli>
<For nr="22.1.1.1.99988">Fornecedores gerais MN</For>
<For nr="22.1.1.2.99988">Fornecedores gerais UE</For>
<For nr="22.1.1.3.99988">Fornecedores gerais OM</For>
<For nr="22.2.1.1.99988">Fornecedores Com Títulos</For>
<For nr="26.1.1.1.99988">Fornecedores Imobilizado MN</For>
</Login>
```

em que:

rc='xx'	Retorno do Erro. 0=Tudo OK, ou: MissingXMLAttr=Faltam os atributos id, seed ou digest. LoginInvalCont=O ID do contacto é inválido ou não existe. LoginInvPasswd=A password do contacto não está correcta.
id='xx'	Devolve o ID do registo de contacto (Autolncremento).
entityID='xxx'	Devolve o ID do registo de terceiro (Autolncremento).
contact='xx.yy'	Devolve o tipo/nº de contacto.
flgCC='1111'	Devolve as 4 flags relativas a C/C, em que o 1º bit representa a 1ª flag,... (janela acessos)
flgEnc='11111'	Devolve as 5 flags relativas a Encomendas em que o 1º bit representa a 1ª flag, ...
flgWeb='0101 ... (16x)'	Devolve as 16 flags específicas para WEB existentes no contacto, 1º bit=1ª flag, ...
Cli nr='21.1...'	Nº que contém informação sobre o nº e tipo de conta do cliente
For nr='22.1...'	Nº que contém informação sobre o nº e tipo de conta do fornecedor

TerFch/SetPass

Este serviço permite modificar a palavra-passe de um contacto de terceiro. Este só pode ser chamado depois de efectuado o ‘TerFch/Login’ com sucesso. O input e output deste serviço são iguais aos do serviço ‘ArtUsr/TestPsw’ (pág. 11).

Este serviço exige que a palavra-passe tenha no mínimo 4 caracteres.

TerFch/Update

Permite actualizar um ou mais registo de terceiros, respectivas moradas extra, contactos e fichas adicionais. Recebe como parâmetro um XML formatado da seguinte forma:

<Entity> <BaseAddr> <!-- fields --> </BaseAddr> →	<OtherAddr> <Rec ID='1'> <!-- fields --> </Rec> <Rec ID='2'> <!-- fields --> </Rec> </OtherAddr> →	<Contacts> <Rec ID='1.1'> <!-- fields --> </Rec> <Rec ID='1.2'> <!-- fields --> </Rec> </Contacts> →	<MoreInfo> <Rec> <!-- fields --> </Rec> <Rec> <!-- fields --> </Rec> </MoreInfo> →
---	--	--	--

A zona ‘fields’ indica que se deve juntar a lista de campos necessários á criação / actualização de um registo, que pode ser consultada no ArtSOFT em Funções de Supervisão/Base de dados/Table Status, na tabela TerFch para ‘BaseAddr’ (endereço base) e na ‘TerCon’ para ‘Contacts’ (contactos). Os campos ‘OtherAddr’ (moradas extra) são um sub-conjunto de TerFch composto por:

NrFilial, Nome, Morada, Localid, CPPais, NrTelef, NrFax, Email, EDICodEnt, LatitudeGPS, LongitudGPS.

O campo ‘NrFilial’ é usado aqui para representar o nº da morada extra a usar, que pode ser expresso de forma alternativa como atributo de <Rec ID='x'>. Se por qualquer razão forem enviados os dois, estes deverão ser iguais, senão é devolvido o erro ‘RecordConflict’.

Para ‘MoreInfo’, os campos disponíveis são:

<MoreInfo id='xx'> <Rec> <RegID>num</RegID> <NrCpo>num</NrCpo> <NmCpo>str: enum</NmCpo> <DataR>date</DataR> <HoraR>hour</HoraR> <NrUserR>num</NrUserR> <UserR>str</UserR> <DataM>date</DataM> <NrUserM>num</NrUserM> →	<UserM>str</UserM> <DataA>date</DataA> <ClassCod>num</ClassCod> <Classif>str: enum</Classif> <Activo>bool</Activo> <Titul>str: subject</Titul> <HLink>str: filepath</HLink> <Texto>str</Texto> <LinkDoc>str: DocID(C F E S Vxxx/yy)</LinkDoc> </Rec> <Rec> <!-- fields --> </Rec> </MoreInfo>
---	--

Se forem enviados o campo ‘RegID’ e o atributo ID='xx’ , estes deverão ser iguais, senão é devolvido o erro ‘RecordConflict’.

TerFch/Delete

Permite eliminar um terceiro, ou informação associada a este, mais especificamente informação de outras moradas, da lista de contactos ou das fichas adicionais.

Input:

```
<Entity id='xx' | EntityID='T:xx.yy'>  
  <BaseAddr type='C|F|All' />  
  <OtherAddr list='w;x;y;z' />  
  <Contacts list='a.a:b.b; c.c:d.d; y.y:z.z' />  
  <MoreInfo ... />  
</Entity>
```

Parâmetros:

id='xx' EntityID='T:xx'	id='xx' - autoincremento do terceiro a eliminar. EntityID='T:xx' - tipo e numero do terceiro a eliminar. No caso de cliente seria na forma ‘C:899’, por exemplo, que significaria que seria para eliminar o cliente com o numero 899.
<BaseAddr type='C F All' />	Se este campo for enviado, o terceiro será eliminado como cliente, fornecedor ou no geral, dependendo do que vem no campo type='...'. Se o campo type='..' não for passado, é assumido o valor por defeito e o terceiro é apagado totalmente (apaga o registo de terceiro). Caso o campo type tenha o valor ‘C’ então apenas o registo de cliente do terceiro é eliminado, mantendo o seu registo de terceiro e de fornecedor (caso o seja). Caso o campo type tenha o valor ‘F’ então apenas o registo de fornecedor do terceiro é eliminado, mantendo o seu registo de terceiro e de cliente (caso o seja). Por último, se no campo type for passado ‘All’ então o registo é eliminado na sua totalidade, ou seja, o registo de terceiro é eliminado.

<OtherAddr list='w;x;y;z' />	Permite eliminar outras moradas, que não a principal, associadas ao registo de terceiro. No campo list='..' devemos passar as moradas que se pretende eliminar. Por exemplo, para eliminar a morada 2 o campo list='..' deverá ser list='2'. Para eliminar todas as moradas extra, o campo list deverá ser list='All'.
<Contacts list='a.a:b.b; c.c:d.d; y.y:z.z' />	Permite eliminar os registos de contactos associados a um terceiro. No caso de o campo list conter list='1,2', significa que o contacto numero 2 do tipo de contacto 1 será eliminado. Se apenas se colocar list='1', então serão eliminados todos os contactos do tipo 1. Tal como outros serviços o ';' e o ':' servem para passar um intervalo de registos em que será aplicada a mesma operação a todos, neste caso a eliminação de registos.
<MoreInfo list='x;y;z' />	Permite eliminar os registos de fichas adicionais associadas ao terceiro. Deverá ser enviado no campo list os autoincrementos das fichas adicionais a eliminar. Se quisermos eliminar mais do que uma ficha adicional devemos enviar uma lista de autoincrementos de fichas adicionais a eliminar, separados por ';'.

TerFch/UpdSug

Permite inserir um pedido de alteração a um registo de um terceiro. Este pedido será entendido pelo ArtSOFT como uma sugestão de alteração, portanto, não a executando de imediato, e deixando essa alteração ao cuidado de um operador que irá validar essas alterações, que, no caso de estarem correctas, serão aceites, senão poderão ser modificadas antes de serem actualizadas ou mesmo descartadas.

O formato da string XML é idêntico ao do TerFch/Update, suportando apenas os nós ‘BaseAddr’ e ‘OtherAddr’, excepto que é obrigatório indicar no nó entity qual o ‘ID’ do terceiro (nº de registo) ou o ‘entityID’ a que as sugestões se referem.

Como apenas alguns campos são passíveis de receberem sugestões de alteração, o protótipo do serviço indica a lista actualizada de campos aceites.

<pre><Entity id='xxx'> <BaseAddr> <!-- fields --> </BaseAddr> →</pre>	<pre><OtherAddr> <Rec ID='1'> <!-- fields --> </Rec> <Rec ID='2'> <!-- fields --> </Rec> </OtherAddr> </Entity></pre>
---	---

Outros serviços similares: ‘GrhEmp/UpdSug’ (pág. 46) e ‘GrhInd/UpdSug’ (pág. 48).

Serviços relativos a Eventos de Terceiros

CRMHdr/GetTable

Fornece uma lista de códigos de eventos, e os respectivos tempos de resposta.

Input:

```
<event codes="2:2.59;3:7;8;9.11:9.19" retid='s' status='S' />
```

event	Nome do nó
codes	Lista / intervalo de códigos a obter, no formato grupo.subgrupo (ex: 1.01:1.49) ou grupo.grupo (ex: 3:5, que é uma abreviatura de 3.00:5.99) ou apenas grupo (ex: 8, que é uma abreviatura de 8.00 a 8.99).

Output:

Event	Nome do nó enviado
nrRecs='xx'	xx= Nº de nós presentes na lista que se segue
Code	Nome do nó com a informação sobre um código de evento
id='xx.yy'	Identificação numérica do código
limit='xxx'	Tempo máximo, em minutos, de resposta do evento. Só presente se xxx>0.
delay=xxxM H D	Tempo de espera em Minutos, Horas, Dias para que o evento comece a ser atendido, e a partir do qual começa a ter efeito o tempo de resposta (limit). Só presente se xxx>0.

Exemplo:

```
<event nrRecs="6">
<code id="2.01">Evento A</code>
<code id="2.02" limit="240">Evento B</code>
<code id="2.03" limit="480">Evento C</code>
<code id="2.04" limit="720">Evento D</code>
<code id="2.05" limit="960" delay="2H">Evento E</code>
<code id="2.06" delay="7D">Evento F</code>
</event>
```

CRMHdr/GetQualif

Fornece uma lista de códigos de qualificadores,e os respectivos nomes de cada um deles.

Input:

```
<root qualif=x:y;z;a:b retid='s' status='S' />
```

root	Nome do nó
qualif	Lista / intervalo de códigos a obter, (ex: 1:2).

Output:

root	Nome do nó enviado
nrRecs='xx'	xx= Nº de nós presentes na lista que se segue
qualif	Nome do nó com a informação sobre um código de qualificador
id='x'	Identificação da tabela de qualificadores
name ='nome_Qualificador'	Nome da tabela de qualificadores pedida
code='xx.yy'	Código do elemento da tabela de qualificadores respectiva.

Exemplo de output:

```
<root nrRecs='4'>
<qualif id='90' name='Versões'>
<rec code='90.01'>Versão 7.50</rec>
<rec code='90.02'>Versão 7.52</rec>
<rec code='90.03'>Versão 7.53</rec>
</qualif>
</root>
```

CRMHdr/Insert

Insere um novo evento.

Input:

```
<event entityID="C:1" ID="2" contact="1.1" [peer='peer_name' ToUser='s\n'] eventlist='1:6'
  code="2.99" class='1.01' startTime='2009-12-31T12:30' tpInput='WEB' tpOutput='WEB'
  retid='s' status='S'>
<subject>Evento de teste das classes XML de eventos</subject>
<message>Este evento serve apenas para teste destas classes</message>
<Qualif Q1='1.1' Q2='2.2' Q8='8.8' />
</event>
```

event	Nome do nó (não é obrigatório que tenha este nome, mas é recomendado)
entityID='nn' ID='xx' contact='xx.yy'	Informação sobre o terceiro a que respeita o evento. Para mais detalhes, ver informações sobre parâmetros gerais dos serviços WEB.
Peer='peer_name'	Identificação do utilizador para quem é dirigido o evento. Se não for definido o evento é enviado para o grupo de atendimento parametrizado na configuração do evento (No ArtSOFT). Para enviar para o utilizador aqui definido tem que se colocar o campo ToUser='s' senão é assumido o valor por defeito, que é 'n' (false), e o evento é enviado para o grupo de atendimento.
ToUser='s\n'	Indicação se é ou não para enviar directamente para um utilizador, definido previamente no campo peer='peer_name'. Por defeito assume o valor 'n'.
code='xx.xx'	Código de evento. Pode ser obtido por 'EvtGetTbl'.
class='xx.xx'	Classificação do evento.
startTime='2009-12-31T12:30'	Data em deve ser registado o inicio do evento. Por omissão, será assumida a data de registo do evento (data de chegada do evento ao servidor de WEB services).
tpInput='xxx'	Meio em que foi recebido o evento. Escolher um elemento da seguinte lista: "phone; fax; mail; email; web; pers". 'pers' deve ser usado para identificação de pedidos feitos pessoalmente.
tpOutput='xxx'	Identico ao anterior, mas identificando o meio usado para a resposta.
subject	O valor deste nó deve conter o assunto.
message	O valor deste nó deve conter a mensagem ou pedido do terceiro.
qualif	Este nó deve conter os atributos de qualificação inicial do pedido, se existirem. Cada qualificador é identificado pelo nome "Q1..Q8", e conter um valor 'xx.yy'.

Output:

```
<event rc="InvalidTercID" />
<event ID="12345" TLimit="2009-04-06T00:09:04"/>
```

event	Nome do nó enviado
id='xx'	Nº do evento inserido (só se foi pedido retid='S')
TLimit='xxx'	Se foi definido um tempo limite para término de execução do evento, este atributo informa a data e hora do término.
rc='xx'	Código de erro ocorrido

CRMHdr/Update

Actualiza um evento já existente.

Input:

```
<event eventID='xxx' contact='x.y' class='x.yy'
  [ tpInput='phone|fax|mail|email|web|pers']
  [tpOutput='phone|fax|mail|email|web|pers']
<subject>subject updated text</subject>
```

```
<message>message text</message>
</event>
```

event	Nome do nó (não é obrigatório que tenha este nome, mas é recomendado)
eventID='nn'	Evento que vai ser actualizado, tem que existir.
contact='x.y'	Contacto de terceiro no caso de se querer alterar o mesmo.
class='xx.xx'	Classificação do evento, caso se queira alterar a classificação do mesmo.
tpInput='xxx'	Meio em que foi recebido o evento (se se quer actualizar ou alterar). Escolher um elemento da seguinte lista: "phone; fax; mail; email; web; pers". 'pers' deve ser usado para identificação de pedidos feitos pessoalmente.
tpOutput='xxx'	Identico ao anterior, mas identificando o meio usado para a resposta.
subject	O valor deste nó deve conter o assunto que se pretende actualizar.
message	O valor deste nó deve conter a mensagem ou pedido do terceiro que se pretende actualizar.

Output:

```
<event id="58"/>
<event rc="InvalidRecNum"/>
```

event	Nome do nó enviado
id='xx'	Nº do evento alterado.
rc='xx'	Código de erro ocorrido.

CRMHdr/Delete

Permite eliminar um evento.

Input:

```
<event eventID='xx' forceDel='s'/>
```

event	Nome do nó enviado
eventID='xx'	Nº do evento a eliminar. Se o evento já tiver tratamento, a operação 'delete' será registada como 'update', a não ser que se envie forceDel='s'.
forceDel='s'	Se forceDel='s' então força a eliminação do evento.

Output:

```
<event id="62" action="D"/>
<event id="58" action="U"/> // caso em que o evento já tinha tratamento.
<event rc="InvalidRecNum"/>
```

event	Nome do nó enviado
id='xx'	Nº do evento eliminado
action="U"	Operação que foi executada, 'U' update e 'D' delete.
rc='xx'	Código de erro ocorrido

CRMHdr/Status

Permite obter uma lista de eventos de determinado terceiro. Recebe como parâmetro um XML formatado da seguinte forma:

Input:

```
<event entityID='x' ID='x' contact='x.x' evtlist='x.x;y.y' code='xx.xx' table='S' detail='F'
after='aaaamdd' maxrec='xx' start='xx' />
```

Parâmetros:

event	Nome do nó
entityID='nn'	Informação sobre o terceiro a que respeita o evento. Para mais detalhes, ver informações sobre parâmetros gerais dos serviços WEB.
contact='ww:xx'	Contacto de terceiro pretendido.
evtList='ww:xx;yy:zz'	Lista de eventos pretendidos. Por omissão, todos. Esta lista pode ser declarada a partir da ficha do contacto, colocando '[EVT='xx.yy;xx.zz]' no respectivo campo 'Observações', permitindo assim definir por contacto, quais os tipos de eventos que costuma ver, ou lhe são permitidos.
code='xx.xx'	Código de evento pretendido. Se existir evtList, code tem que ser um dos elementos dessa lista, senão é assinalado 'NotAllowedEvt';
table='S N'	Pretende-se que seja devolvida a tabela com os eventos usados.
detail='F P A'	Se 'F' inclui apenas eventos já finalizados, se 'P' inclui apenas eventos pendentes, se 'A' inclui todos os eventos.
after='2009-12-31T12:30'	Data em deve ser registado o inicio do evento. Por omissão, será assumida a data de registo do evento (data de chegada do evento ao servidor de WEB services).
maxrec='xxx'	Indica que devem ser devolvidos apenas 'xxx' registos. Se não for especificado, assume por omissão 100.
start='xx'	Apenas a partir do evento com ID='xx' (para continuação de query anterior que excedeu 'maxrec' registos.

Output:

event	Nome do nó enviado
Table	Nome do nó que contém a tabela de designações dos eventos
code id='xx'	Código de Evento
records	Nome do nó que contém a lista de eventos pedidos
evt id='xxx'	Id do evento
code='xx'	Código do evento
status='x'	Estado do evento (pendente, finalizado)
date='xx'	Data e hora de inicio do evento
limit='xx'	Data limite de resolução do evento
subj='xx'	Texto do assunto (se existir)
text='xx'	Texto (se existir)
updates='xx'	Textos de alteração ao pedido original (se existir)
answer='xx'	Resposta ao terceiro (se existir)

Exemplo:

```

<event rc='xxxxx' /> ← código de erro
ou
<event> ← Lista válida
<table nrRecs="4">
  <code id="1.01">Pedido de Proposta</code>
  <code id="2.02">Pedido de assistência</code>
  <code id="9.09">Reclamação</code>
</table>
<records nrRecs="3">
  <evt id="47670" code="1.01" status="P" date="2099-01-04T09:57:58">
    <subj>Proposta de Software</subj>
    <text>Proposta para ERP para Sede + 5 Filiais, sendo uma em Espanha e outra na Madeira</text>
  </evt>
  <evt id="47656" code="2.02" status="P" date="2099-01-04T09:40:00">
    <subj>Assistência para configuração de contas email</subj>
  </evt>
  <evt id="47594" code="9.09" status="P" date="2099-12-31T11:04:12">
    <subj>Insatisfação total</subj>
  </evt>

```

CRMHdr/GrupTask

(em construção)

CRMHdr/PersTask

(em construção)

CRMHdr/FollowUp

(em construção)

CRMHdr/GetImage

Fornece a(s) imagem(s) associadas a um evento. O formato de chamada e retorno são iguais aos de *DocFch/GetImag*, excepto a identificação do evento só pode ser feita por ‘id=’xx’, uma vez que a opção ‘nrDoc=x/y’ aqui não faz sentido.

CRMHdr/SetImage

Permite associar imagem(s) a um evento. O formato de chamada e retorno são iguais aos de *DocFch/SetImag*, excepto a identificação do evento só pode ser feita por ‘id=’xx’, uma vez que a opção ‘nrDoc=x/y’ aqui não faz sentido.

VndFch/Update

(em construção)

Serviços relativos a Contabilidade e Contas Correntes

Estes serviços estão subdivididos em duas categorias: inserção / actualização da base de dados e extractos especializados, com um conjunto de informação frequentemente utilizada, que, se for pretendida de outra forma, ou com outros dados, poderá sempre ser obtida via ‘Query’.

CtaLan/CurrAcc

Elabora um extracto de conta de todos os movimentos das contas solicitadas.

Input:

Parâmetro	Descrição
accounts='xxx;yyy;zzz'	Lista de contas ou lista de intervalos de contas (obrigatório)
code='A1:A999;B1:B999'	Lista de tipos de documento ou lista de intervalos de tipos de documento
nrdoc='xxx'	Nº do documento
all='s'	True=Todos os lançamentos, False=Apenas documentos com saldo em aberto
after='date'	Lançamentos com data igual ou superior a ‘date’
before='date'	Lançamentos com data igual ou inferior a ‘date’
defs='s'	Enviar tabela de definições de códigos
maxrecs='xx'	Enviar no máximo, ‘xx’ registos. Este limite não pode ser superior a 100.

Exemplo:

```
<tag account='21.1.1.001;21.2.001' code='A1:A19;A58:A69' nrdoc='123' all='S'
      after='01.01.2019' before='31.12.2019' defs='s' maxrecs='25' status='n' retid='s' />
```

Output:

```
<CurrAcc>
<Table NrRecs="3">
<Code id="A012">N/Nota de Débito</Code>
<Code id="A502">Aceite Letras BPI</Code>
<Code id="A503">Aceite Letras BCP</Code>
</Table>
<Records NrAccs="2" TtRecs="7">
<Account Nr="211114781" NrRecs="4">
<Rec Code="A012" Docum="2307" Date="2008-12-31" Value="118.21" Balance="118.21" Refer="V005/2307"/>
<Rec Code="A012" Docum="2308" Date="2008-12-31" Value="166.22" Balance="166.22" Refer="V005/2308"/>
<Rec Code="A012" Docum="2309" Date="2009-01-15" Value="85.72" Balance="85.72" Refer="V005/2309"/>
<Rec Code="A012" Docum="2315" Date="2009-02-06" Value="107.43" Refer="V005/2315"/>commentários</rec>
</Account>
<Account Nr="21214781" NrRecs="3">
<Rec Code="A503" Docum="1246" Date="2008-03-04" DueDate="2008-05-30" Value="9300.06" Balance="9300.06"/>
<Rec Code="A503" Docum="1262" Date="2008-12-19" DueDate="2009-04-15" Value="4674.04" Balance="4674.04"/>
<Rec Code="A502" Docum="1260" Date="2008-12-19" DueDate="2009-03-16" Value="4674.06" Balance="4674.06"/>
</Account>
</Records>
</CurrAcc>
```

CtaLan/OpenDoc

Elabora um extracto de conta com os movimentos de documentos não saldados (ainda em aberto).

Output:

```
<OpenDoc>
...
</OpenDoc>
```

O resultado deste extracto é idêntico ao do anterior, excepto que o nome ‘CurrAcc’ passa agora para ‘OpenDoc’.

CtaLan/DocType

Elabora um extracto de conta com os movimentos de documentos do(s) tipo(s) indicados.

Output:

```
<DocType>
  ...
</DocType>
```

CtaLan/DocRecv

Elabora um extracto de conta com os movimentos de documentos do(s) tipo(s) indicados, e respectivos pagamentos (a fornecedores) / recebimentos (de clientes).

Output:

```
<DocRecv>
<Records NrAccs="1" TtRecs="2">
  <Account Nr="211114777" NrRecs="2">
    <Rec Code="A001" Docum="12153" Date="2008-11-13" Value="800.75" Balance="400.75" Refer="V001/12153">
      <OnAcc Code="B032" Docum="59" Date="2009-01-28" Value="575.45" Remitt="200" Refer="TB"/>
      <OnAcc Code="B032" Docum="91" Date="2009-02-28" Value="255.15" Remitt="200" Refer="TB"/>
    </Rec>
    <Rec Code="A030" Docum="97" Date="2009-02-27" Taxes="23.2" Value="13.2" Balance="9.2" Refer="V012/7"/>
  </Account>
</Records>
</DocRecv>
```

CtaLan/RecvDoc

Elabora um extracto de conta com os movimentos de pagamentos de documentos do(s) tipo(s) indicados, e os respectivos regularizados, ou seja, é o inverso do extracto 'DocRecv'.

Output:

```
<DocRecv>
  ...
</DocRecv>
```

Descrição dos nós e atributos de output de todos os tipos de extractos:

Nó / Atributo	Descrição
currAcc	Nome do extracto
Table	Solicitado por defs='s', contém o código e respectiva descrição
Records	Nó que contém a lista de contas
ttRecs / nrRecs	Nº total de movimentos presentes de todas as contas ('ttRecs'), ou só numa conta ('nrRecs'). Se seguido de '+' o extracto está incompleto, porque ultrapassa o nº de registos definidos por 'maxrecs=xx'.
Account	Nó que conterá o extracto da conta 'nr', com o total de 'nrRecs' movimentos
Rec	Nó que representa um movimento de uma conta.
code	Código deste movimento
docum	Nº do documento
date	Data valor do movimento
dueDate	Data de vencimento do movimento, se existir
value	Valor do movimento
balance	Valor em aberto do movimento, ou o que ainda falta pagar / receber
refer	Documento de referência, se existir
pend	O movimento ainda está pendente de aprovação
currency	Moeda em que foi feito o movimento
taxes	Valor de impostos incluídos no valor total do movimento
delta	Desconto deduzido ou agravamento incluído no total do movimento
remitts	Num lançamento de débitos de clientes / créditos de fornecedores informa que existem 'n' pagamentos associados, que irão ser descritos a seguir
remitt	Nos extractos 'LanDocRecv' e 'LanRecvDoc', informa qual o valor usado para regularização do movimento
dbAnt	Débito anterior á data do extracto

crAnt	Crébito anterior á data do extracto
onAcc	Nó que representa uma regularização do documento ao qual está dependente (On Account, por conta)

CtaLan/GetImage

Fornece a(s) imagem(s) associadas a um documento de C/Correntes. O formato de chamada e retorno são iguais aos de *DocFch/GetImag*.

CtaLan/SetImage

Permite associar imagem(s) a um documento de C/Correntes. O formato de chamada e retorno são iguais aos de *DocFch/SetImag*.

CtaLan/LanDelete

Permite eliminar lançamentos de documentos contabilísticos.

Para eliminar todos os lançamentos de um documento:

```
<docum DocID='xxx/NrDoc' [retID='Y|N'] [status='Y|N'] [trans='Y|N'] />
```

docum	Nome (pode usar-se qualquer um) do nó root.
DocID='xxx/yyy'	Qual o tipo do documento (xxx) e qual o documento (yyy) a eliminar.
[retID='Y N'] [status='Y N'] [trans='Y N']	Descrito em Parâmetros gerais dos Serviços WEB

Para eliminar todos os lançamentos de vários documentos:

```
<docum TpDoc='xxx' NrDocI='y' NrDocF='z' [retID='Y|N'] [status='Y|N'] [trans='Y|N'] />
```

docum	Nome (pode usar-se qualquer um) do nó root.
TpDoc='xxx'	Qual o tipo do documento (xxx) do qual se pretende eliminar vários documentos.
NrDocI='y'	Número do documento inicial (y) a ser eliminado. Por exemplo se pretendermos apagar do documento 2 ao 10, y será 2 e z será 10, sendo que serão eliminados todos os documentos nesse intervalo.
NrDocF='z'	Número do documento final (z) a ser eliminado.
[retID='Y N'] [status='Y N'] [trans='Y N']	Descrito em Parâmetros gerais dos Serviços WEB

CtaLan/LanUpdate

Permite inserir lançamentos contabilísticos, e eventuais resgistros associados, como por exemplo contas da contabilidade.

Para inserir lançamentos contabilísticos e contas associadas (opção 1):

```
<message trans='S' retid='S' status='S'>
  <account>
    - fields - os mesmos elementos da mensagem 'CtaFch/UpdateAccounts' -
  </account>
  <account>
    - fields -
  </account>
<transactions>
  <journal id='x.y' [trans='x']>
    <docum [docID='tpDoc/nrDoc'] date='xxxx' refdate='xx' [trans='x']>
      <entry>
        - fields -
      </entry>
      <entry>
        - fields -
      </entry>
    </docum>
    <docum [id='rec_nr' | docID='tpDoc/nrDoc'] [trans='x']>
      ...
    </docum>
  </journal>
  <journal id='x.y' [trans='x']>
    ...
  </journal>
</transactions>
</message>
```

message	Nome (pode usar-se qualquer um) do nó root.
retID='Y N' status='Y N' trans='Y N'	Descrito em Parâmetros gerais dos Serviços WEB

<account> - fields - </account>	Os mesmos elementos da mensagem ‘ CtaFch/UpdateAccounts ‘
<transactions>...</transactions>	Início da secção de lançamentos contabilísticos a inserir.
<journal id='x.y' [trans='x']> ...</journal>	Secção respeitante ao diário a ser inserido. Id='x.y' refere-se ao número do diário a inserir. Podem ser inseridos vários diários, desde que estão dentro do nó <transactions>.
<docum [docID='tpDoc/nrDoc'] date='xxx' refdate='xx' [trans='x']> ...</docum>	Documento a ser inserido dentro do diário. Para inserir é necessário passar qual o tipo de e numero do documento a ser inserido (docID='tpDoc/nrDoc'). Podem ser inseridos vários documentos, desde que dentro de um nó <journal>.
<entry> - fields - </entry>	Dados do lançamento de documento a ser inserido. Fields corresponde a campos da tabela CtaLan. Podem ser inseridos vários lançamentos, desde que dentro de um nó <docum>.

Para inserir apenas lançamentos contabilísticos (opção 2):

```
<message trans='S' retid='S' status='S'>
<transactions>
  ...
</transactions>
</message>
```

message retID='Y N' status='Y N' trans='Y N'	Nome (pode usar-se qualquer um) do nó root. Descrito em Parâmetros gerais dos Serviços WEB
<transactions> ...</transactions>	Início da secção de lançamentos contabilísticos a inserir. Terá que ter os mesmos nós “filhos” que foram descritos na “opção 1” , <journal> , <docum> e <entry> .

CtaFch/UpdateAccounts

Permite inserir ou actualizar contas no plano de contas.

Para inserir contas :

```
<account id='Ai' retid='S|N|T|F' trans='S|N|T|F' status='S|N|T|F'>
  - fields -
</account>
```

account id='Ai'	Nome (pode usar-se qualquer um) do nó root. Auto-incremento da conta. Se este for passado irá fazer update a uma conta já existente ou dar erro caso a conta não exista. Se se pretende inserir uma conta nova, então este campo não deve ser passado, sendo que no espaço fields devemos passar o campo NrConta ou o campo NrContaEd obrigatoriamente.
retid='S N T F' trans='S N T F' status='S N T F'> - fields -	Descrito em Parâmetros gerais dos Serviços WEB No espaço fields, devem ser passados, os campos da tabela CtaFch, que se pretende inserir ou actualizar relativamente a uma conta. Os campos TipoCta e (NrConta ou NrContaEd) pertencentes à tabela CtaFch têm que obrigatoriamente ser passados quando se está a inserir uma nova conta. No caso de se estar a realizar a actualização de uma conta e o id='Ai' tenha sido passado, então apenas o campo TipoCta tem obrigatoriedade de ser passado.

CtaFch/DeleteAccounts

Permite eliminar contas do plano de contas.

No caso de contas de Terceiros, estas apenas podem ser apagados através da eliminação do terceiro .

Para eliminar contas:

```
<account id='Ai' | account='12.1.1' retid='S|N|T|F' trans='S|N|T|F' status='S|N|T|F' />
```

account	Nome (pode usar-se qualquer um) do nó root.
id='Ai'	Auto-incremento da conta a ser eliminada. Deve ser passado ou id='Ai' ou account='xx.y.z', caso sejam passados ambos, o serviço irá ter em conta apenas id='Ai'.
account='12.1.1'	Número da conta a ser eliminada.
retid='S N T F' trans='S N T F' status='S N T F'	Descrito em Parâmetros gerais dos Serviços WEB

Serviços relativos a Recursos Humanos

GrhEmp/Login

Efectua o login de um empregado no servidor XML ArtSOFT afim de poder obter acesso aos seus dados como empregado. Para isso, o ArtSOFT apresenta o código de identificação de cada empregado na respectiva ficha, onde, nesse mesmo sítio, permite atribuir uma palavra-passe gerada aleatoriamente, que deverá depois ser alterada pelo próprio empregado, já no interface externo (WEB ou outro).

Deve receber um XML formatado da seguinte forma:

Input:

```
<Login id='LBW0XGXL' seed='0123456789012345' digest='FEDBD236CCC8D897947EC9B7B3AD46667947EC' />
```

- O atributo ‘ID’ deve conter o identificador do registo de empregado, fornecido pelo ArtSOFT na respectiva ficha, a partir do separador ‘diversos’. Este parâmetro serve para localizar qual o seu registo de empregado, e é calculado pelo ArtSOFT, de forma a ter sempre 8 dígitos entre ‘0..9’ e ‘A..Z’ (não diferenciando entre maiúsculas e minúsculas), e ser ‘pseudo-aleatório’, de forma a não poder ser facilmente dedutível qual o ID de um empregado a partir do ID de outro, afim de aumentar a segurança do sistema.
- Os atributos ‘seed’ e ‘digest’ e a descrição do algoritmo de login estão descritos no [Apêndice K - Validação de Logins](#).

Output:

```
<Login rc='xx'>Nome do Empregado ou Independente</Login>
```

GrhEmp/SetPass

Este serviço permite alterar a palavra-passe de empregado para acesso aos seus dados de empregado no ArtSOFT. Para alterar a password é necessário ter login válido, pois só assim é que se pode validar qual o empregado que está a mudar a password.

Deve receber um XML formatado da seguinte forma:

Input:

```
<root psw='xyz/123?ABC!456' />
```

- O atributo ‘psw’ deve conter a password que se pretende associar ao empregado.

Output:

```
<root/> //Quando o set da password foi feito com sucesso  
<root rc='NotLoggedIn|PasswTooShort|PasswTooWeak' /> //Quando o set da password não foi feito com sucesso
```

GrhEmp/Update

Este serviço permite inserir ou actualizar os dados da ficha de um empregado, do seu agregado familiar ou da sua ficha adicional.

Input:

```
<employee id='nrEmp' .../>
<mainrec>
  - fields -
</mainrec>
<family>
  <rec>
    - fields -
  </rec>
  <rec> ... </rec>
</family>

</moreInfo>
<rec>
  - fields -
</rec>
<rec>...</rec>
</moreInfo>
</employee>
```

Secção do XML	Obrig	Descrição
<employee id='nrEmp' .../>	S	Início da mensagem e eventuais atributos adicionais que sejam necessários
<MainRec> - fields - </ MainRec >	N	Secção opcional para a identificação de um empregado que exista (no caso de se estar a inserir) ou não (no caso de se tratar de uma actualização). Se esta secção não existir, o atributo id='nrEmp' tem que estar presente no nó employee.
<family> <rec> - fields - </rec> <rec> ... </rec> </family>	N	Secção opcional para a informação relativa ao agregado familiar de um empregado.
</moreInfo> <rec> - fields - </rec> <rec>...</rec> </moreInfo>	N	Secção opcional para a informação relativa às fichas adicionais de um empregado.
</employee>		

Output:

...

GrhEmp/UpdSug

Permite inserir um pedido de alteração a um registo de um empregado. Este pedido será entendido pelo ArtSOFT como uma sugestão de alteração, portanto, não a executando de imediato, e deixando essa alteração ao cuidado de um operador da área de recursos humanos que irá validar essas alterações eventualmente contra documentos a apresentar, que, no caso de estarem correctas, serão aceites, senão poderão ser modificadas antes de serem actualizadas ou mesmo descartadas. O seu funcionamento é idêntico ao de ‘TerFch/UpdSug’ (pág. 33), não existindo o nó ‘OtherAddr’, e aceitando uma lista de campos diferente.

Input:

```
<employee id='x' .../>
  - fields: -
</employee>
```

Secção do XML	Obrig	Descrição
<employee id='x' .../> - fields: -	S	Início da mensagem e eventuais atributos adicionais que sejam necessários.

</employee>	No espaço fields iremos colocar os campos do empregado que irão ser sugeridos para alteração.
-------------	---

GrhEmp/UpdEvent

Este serviço permite inserir eventos para processamento de remunerações, horas extra, férias, faltas ou abonos/descontos particulares para um empregado. Devido ao polimorfismo desta tabela, descrevem-se detalhadamente as várias colunas

Campo	I/O	Descrição
AI_Evp	R	Autolncremento do registo
NrEmp	R/W	Nº do empregado
DataEf	R/W	Data a partir da qual se pode processar, ou data inicial das faltas
DataVal	R/W	Data a que respeita a remuneração (se respeitante ao mes corrente = 0) (Ex. retroactivos, faltas de meses anteriores não processadas, etc)
DataReg	R/W	Data em que foi registado o evento (só para fins informativos)
DataProc	R	Data em que este evento foi processado.
Evento	R/W	Falta ou remuneração (vencimento, horas extra/trab.nocturno, em feriados, dias de descanso, etc)
NaoProc	R	Flag 'ainda não processado'
HrMinIni	R/W	Hora/Minuto inicio do evento
TipoUnid	R/W	Unidade base do evento: -Remunerações: 0=RemunMensal, 1=RemunUnitar, 2=RemunHoraria, 3=RemunEventual; -Faltas: 0=Falta ao Minuto, 1=FaltaAoDia
NrUnidad		Nº de eventos repetidos (não podem ser interpolados que nesse caso, gera n registos)
Valor	R/W	Valor da remuneração ou Nº de Minutos físicos. - Horas extra: Não pode registar mais que o máximo legal por dia. - Faltas: O dia inicial+nº Horas não pode ultrapassar a data de processamento da empresa, ou fim do ano.
LocTrab	R/W	Código do Local de Trabalho
AI_Cad	R/W	FK: Ligação a evento de cadastro
AI Grupo	R/W	FK: Autolncremento do Evento que agrupa um conjunto de eventos da mesma natureza
RefRemFix	R	Referencias as remunerações fixas (usadas nos retroactivos automáticos)
Obs [100]	R/W	Este texto será transportado para as observações do processamento

GrhEmp/DelEvent

Este serviço permite eliminar um evento de processamento de remunerações, horas extra, férias, faltas ou abonos/descontos particulares de um empregado.

Input:

```
<employee id='nrEvent' />
```

GrhEmp/UpdEventCad

Este serviço permite inserir eventos de cadastro de empregados.

Input:

```
<employee id='1' retID='s' status='s' trans='s'
      <rec>
        - fields (GrhCad) -
      </rec>
</employee>
```

GrhEmp/DelEventCad

Este serviço permite eliminar eventos de cadastro de empregados.

Input:

```
<employee id='nrEvent' />
```

GrhInd/Login

Efectua o login de um colaborador independente no servidor XML ArtSOFT afim de poder obter acesso aos seus dados como independente. Para isso, o ArtSOFT apresenta o código de identificação de cada independente na respectiva ficha, onde, nesse mesmo sítio, permite atribuir uma palavra-passe gerada aleatoriamente, que deverá depois ser alterada pelo próprio empregado, já no interface externo (WEB ou outro). Os parâmetros que recebe e os resultados que fornece são idênticos aos do serviço 'GrhEmp/Login'.

GrhInd/SetPass

Este serviço permite alterar a palavra-passe de um colaborador independente para acesso aos seus dados no ArtSOFT. O input e output deste serviço são iguais aos do serviço 'ArtUsr/TestPsw' (pág. 11). Este serviço exige que a palavra-passe tenha no mínimo 8 caracteres.

GrhInd/Update

Este serviço permite inserir ou actualizar os dados da ficha de independente, ou da sua ficha adicional.

GrhInd/UpdSug

Permite inserir um pedido de alteração a um registo de um independente. Este pedido será entendido pelo ArtSOFT como uma sugestão de alteração, portanto, não a executando de imediato, e deixando essa alteração ao cuidado de um operador da área de recursos humanos que irá validar essas alterações eventualmente contra documentos a apresentar, que, no caso de estarem correctas, serão aceites, senão poderão ser modificadas antes de serem actualizadas ou mesmo descartadas. O seu funcionamento é idêntico ao de 'GrhEmp/UpdSug' (pág. 46), diferindo apenas na lista de campos aceites.

GrhInd/UpdEvent

Este serviço permite inserir eventos para processamento de remunerações para independentes. De seguida apresentam-se os nomes e a descrição de cada coluna desta tabela.

Campo	I/O	Descrição
NrIndep	R/W	N.º de independente
CodRemum	R/W	Código de remuneração
NumSeq	R/W	Nº de Sequência (para permitir vários CodRemun para o mesma data)
DataEvt	R/W	Data do evento
DataVal	R/W	Data a que respeita o valor
DataPro	R	Data de processamento
Process	R	Flag 'processado'
Local	R/W	Código de Local de trabalho
Quant	R/W	Quantidade
Valor	R/W	Valor
NrHoras	R/W	Nº de horas, quando a remuneração for utilizada para o anexo F do relatório único
Obs[127]	R/W	Observações

Apêndice A - XML Reports

Um relatório do tipo ‘lista’ começa com o atributo **type='list'**, seguido do atributo **query** que definirá que tipo de registos se pretende. É ainda possível definir que nomes terão os nós que representam as linhas, com **name='nome#xxx'**, que, se não for definido, assumirá ‘Row’. xx representa o nº de dígitos que completará o nome, e simultaneamente define o nº máximo de registos devolvidos ($10^{\text{xxx}-1}$), ou seja, para xxx=3, será $10^{3-1}=999$. Este valor pode ainda ser definido pelo atributo **max='yyy'**, sendo ‘yyy’ o nº máximo pretendido. Se definidos os dois, ‘yyy’ não deve ser superior a ‘xxx’.

A seguir, é necessário definir as colunas do relatório, através de ‘defcol’.

...

Exemplo 1

```
<TBL type='list' query='StkFch|Principal|Codigo=1010123;1010128;1010145' name='SKU#2'>
<defcol>
  <Codigo form='%StkFch.Cod.Editado' type='text' />
  <Nome form='%StkFch.Nome.0' type='text' />
  <PrecoV form='%StkFch.Preco.0' />
  <NmImgS form='%StkFch.Div.FichImg' type='text' />
  <NmImgR form='%StkFch.Div.FichImgRed' type='text' />
  <ImgStd form='$GetImage(%StkFch.Div.FichImg)' type='text' />
  <ImgRed form='$GetImage(%StkFch.Div.FichImgRed)' type='text' />
</defcol>
</TBL>
```

Output:

```
<TBL>
<SKU01>
  <Codigo>1.01.0123</Codigo>
  <Nome>Computador 99.0GHz, 2Tb RAM, 999TB Disco</Nome>
  <PrecoV>1672</PrecoV>
  <NmImgR>{ART}\Imagem1.png</NmImgR>
  <ImgRed>iVBORw0KGgoAAAANSUhEUgAAAV8AAAFbCAIAAAD9RhrAAAAAB3RJTUUH1QEFFFgQJgYJc3AAAHB1J
    REFUeJztnbt1IzsShjF7ZNC8Aaw5Qawp4wYwQWwQMmSUMYaCuEEogDVkrDFBjHkDWJMGjTWaoof4
    ...
    WxzYAUYNPI/M38SOv7ldT9gBTIrvi1tTwmKBWTsLBnkHMCONIojE1r65FE4QOwAA3GBNagCAG9gB
    AOAGdgAAuIEdAAbu/g8etvGEfkhiXAAAABJRUSErkJggg==</ImgRed>
</SKU01>
<SKU02>
  <Codigo>1.01.0128</Codigo>
  <Nome>Computador 99.66GHz, 2Tb RAM 999TB</Nome>
  <PrecoV>1912.5</PrecoV>
  <NmImgR>{ART}\Imagem2.gif</NmImgR>
  <ImgRed>R01GODlhqgCmApcAAB8aF0RAPkVBPkpGRFpWVGlmZod5Gd+IPd+MQ96RTt2QTNuTVNyZxtiTWOKN
    Q+SSS+yWS+yWTOyXTumSeYyT+aRR+qZUeyZue2aUu2dVu2cVeubVeSaWu2fWu2eWeqeXoazVe6h
    Xt2dZdWdbNicaOSfytsGc9yjc9eme9aide6jYe61Zu2jYuSkbO6naemla++oau+qbemobuOgZvCr
    ...
    BmqohmvQhm84h/d0B9UMnOb8USlJ8zzQswfRczt0OAdz+AbwM8ETbNMQ1IZt+AzzOAd0WId24FIM
    9Viw3VN1+jy2i1N4wD7tcyvtwz54qAd7uIceFRw+bVRHtzSAAA7</ImgRed>
</SKU02>
<SKU03>
  <Codigo>1.01.0145</Codigo>
  <Nome>Computador 99.99GHz, 2Tb RAM, 999TB</Nome>
  <PrecoV>2102.15</PrecoV>
  <NmImgR>{ART}\Imagem3.gif</NmImgR>
  <ImgRed>R01GODlhhpCcoAPcAAB8aFzMuKzUwLjolMzs3NEI900RAPU1FQk9LSFFNS1RRT11VU1xYV19bWVdT
    UWJeXGRgXmViYGlmZG5raWxoZnFubHRxb3ZcX12dHt4dnx5d316eH57eX98eud5GYB9e4F/fYF+
    ...
    00Vv/uconVJMZYcFRIdx8FRvN8VWhc7WbvCGcXBUDWCHSq0HOO7nXf7nnQ1oca6HeLDUTJXUSA3U
    d4ghir6HbvZnjE5plR71gAAAow==</ImgRed>
</SKU03>
</TBL>
```

Funções internas de XMLReports

Retorno	Nome da função	Descrição
Funções de Datas		
num	\$julian(string source) \$julian(enum relativos)	Converte uma data gregoriana (normal) 'source' para uma data juliana ⁶ . -Subtraindo duas datas julianas, obtém-se o nº de dias entre as duas datas; -Porque as datas julianas são lineares, é possível usar as operações de adição e subtracção com elas, convertendo-se depois o resultado uma data gregoriana (normal). -Tendo uma data juliana, o resto da sua divisão por 7 (módulo 7), fornece o dia da semana (com Segunda=0 .. Domingo=6). Ex: \$Mod(\$Julian(2000-01-01),7) = 5 (Sábado). -Para a alternativa 'relativos', ver tabela abaixo.
str	\$julToDate(int numero, [fmtDatas format])	Transforma 'numero' (uma data em formato Julian) para o formato usual, de acordo com 'format'. Se format não for definido, assume 'AAAAMMDD'.
str	\$date(string source, [fmtDatas format]) \$date(enum relativos, [fmtDatas format])	Transforma uma data 'source' num qualquer formato usual, para o formato tipo 'format'. Se format não for definido, assume 'AAAAMMDD'. -Para a alternativa 'relativos', ver tabela abaixo.

Funções de Horas		
str num	\$hour(string source, [fmtHoras format]) \$hour(enum relativos, [fmtHoras format])	Transforma uma hora em formato 'source' para o formato 'format'. Tendo uma hora em formato segundos (ou seja, um valor numérico simples), é possível fazer cálculos de tempos expressos em segundos . Atenção: o formato 'Miliseg' não deve ser usado para inputs de cálculos de tempos, a não ser que se divida o mesmo por 1000. Se format não for definido, assume 'HH:MM:SS'. Se format='Seg' ou 'Miliseg' devolve 'Num' senão, devolve 'Str' A opção relativos só tem um parâmetro ('Now'), que devolve a hora do sistema.

Formatos de Datas:

fmtDatas	fmtDatas (alt)	Resultado
DD/MM/AAAA	D/M/A	31/12/2999
AAAA/MM/DD	A/M/D	2999/12/31
DD.MM.AAAA	D.M.A	31.12.2999
AAAA.MM.DD	A.M.D	2999.12.31
DD MMM AAAA	D M A	31 Dez 2999
AAAA MMM DD	A M D	2999 Dez 31
DDMMMAAAA	DDDDMA	31Dez2999
AAAMMMDD	AMMD	2009Dez31
DDMMAAA	DMA	31012999
AAAMMDD	AMD	29991231
DATAEXT	DX	Sexta, 31 de Março de 2999

Formatos de Horas:

fmtHoras	fmtHoras (alt)	Resultado
HH:MM:SS	H:M:S	23:59:59
HH:MM	H:M	23:59
HH:MM:SS.MIL	H:M:S.M	23:59:59.987
HhMmSs	HMS	23h 59m 59s
HhMm	HM	23h 59m
HhMmSsMil	HMSM	23h 59m 59s 987ms
DHhMmSs	DHMS	65d 23h 59m 59s
DHhMm	DHM	65d 23h 59m
Seg		999999999
Miliseg		999999999999

Parâmetros relativos de Datas

relativos	Resultado
Today	Data actual (de sistema)
BegYear	1 de Janeiro do ano actual
EndYear	31 de Dezembro do ano actual
BegMonth	1 do mês e ano actuais
EndMonth	Último dia do mês e ano actuais
Minus1Y	Data actual menos um ano
Minus6M	Data actual menos 6 meses
Minus3M	Data actual menos 3 meses
Minus2M	Data actual menos 2 meses
Minus1M	Data actual menos 1 mês
Minus3W	Data actual menos 3 semanas

Parâmetros relativos de Datas (Cont.)

relativos	Resultado
Minus2W	Data actual menos 2 semanas
Minus1W	Data actual menos 1 semana
Plus1Y	Data actual mais 1 ano
Plus6M	Data actual mais 6 meses
Plus3M	Data actual mais 3 meses
Plus2M	Data actual mais 2 meses
Plus1M	Data actual mais 1 mês
Plus3W	Data actual mais 3 semanas
Plus2W	Data actual mais 2 semanas
Plus1W	Data actual mais 1 semana

Funções Lógicas

bool	\$isEqual(undefined expr1, undefined expr2)	Verifica a igualdade entre 'expr1' e 'expr2', e devolve '1' (true) se forem iguais, ou '0' (false) no caso contrário.
bool	\$isEqual(undefined expr2)	Verifica a igualdade entre o valor da célula onde se encontra a fórmula (que funciona como 'expr1'), e 'expr2'. Devolve '1' (true) se forem iguais, senão '0' (false).
bool	\$isNotEq(undefined expr1, undefined expr2)	Verifica a desigualdade entre 'expr1' e 'expr2', e devolve '1' (true) se forem diferentes, ou '0' (false) no caso contrário
bool	\$isNotEq(undefined expr2)	Verifica a desigualdade entre o valor da célula e 'expr2', e devolve '1' (true) se forem diferentes, senão '0' (false).
bool	\$isGreat(undefined expr1, undefined expr2)	Verifica se 'expr1' é maior que 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isGreat(undefined expr2)	Verifica se o valor da célula é maior que 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isGreatEq(undefined expr1, undefined expr2)	Verifica se 'expr1' é maior ou igual a 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.

⁶ Uma data juliana é um número de dias sequencial, logo, possível de ser utilizado para efectuar cálculos de datas. No exemplo: Julian(20190101)+45, adiciona 45 dias ao dia 1 de Janeiro, sendo o resultado o equivalente juliano ao dia 15 de fevereiro. Para obter o resultado em data usual (Gregoriana), é necessário convertê-la com JulDate. Ex: \$JulDate(Julian(20190101)+45).

bool	\$isGreaterEq(undef expr2)	Verifica se o valor da célula é maior ou igual a 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isLess(undef expr1, undef expr2)	Verifica se 'expr1' é menor que 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isLess(undef expr2)	Verifica se 'expr1' é menor que 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isLessEq(undef expr1, undef expr2)	Verifica se 'expr1' é menor ou igual a 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$isLessEq(undef expr2)	Verifica se o valor da célula é menor ou igual a 'expr2', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
bool	\$inRange(undef value, undef limInf, undef limSup)	Verifica se 'value' é maior ou igual a 'limInf' e menor ou igual a 'limSup', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário
bool	\$outRange(undef value, undef limInf, undef limSup)	Verifica se 'value' é menor que 'limInf' ou maior que 'limSup', e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário
bool	\$isTrue(string logicExpr)	Avalia uma expressão lógica, e devolve o resultado: 0/1 (true/false). 'logicExpr' pode conter uma série de expressões ligadas por '^' (AND) ou ' ' (OR), podendo ser antecedidas por '-' (NOT). Exemplo: "(-val1 ^ val2 val3) -(val4 ^ val5)"
bool	\$logicalAND(uint32 flags, string bitList, char baseNum)	Verifica em 'flags' se a lista de bits (1..64) indicada em 'bitList' está toda 'on'. 'flags' é um numérico em formato decimal ou hexadecimal; 'bitList' é uma lista de números separada por ';' e 'baseNum' é um caractere 'D' (Decimal) ou 'H' (Hexadecimal) indicando qual a base numérica utilizada. Devolve o resultado: 0/1.
Bool	\$logicalAND(uint32 flags, string bitList, char baseNum)	Verifica em 'flags' se na lista de bits (1..64) indicada em 'bitList' pelo menos um deles está 'on'. 'flags' é um numérico em formato decimal ou hexadecimal; 'bitList' é uma lista de números separada por ';' e 'baseNum' é um caractere 'D' (Decimal) ou 'H' (Hexadecimal) indicando qual a base numérica utilizada. Devolve o resultado: 0/1.
bool	\$exists(string cellAddr)	Verifica se a célula 'cellAddr' existe e contém valor não vazio, e devolve 1 (true), ou 0 (false) caso contrário. 'cellAddr' deve conter o caminho absoluto (root/node1/subn), ou relativo (.../node/subn) do nó XML pretendido.

Funções Numéricas

Num	\$abs(double val)	Devolve o valor absoluto de 'val' (quer o valor seja positivo ou negativo, devolve sempre o seu valor positivo).
Num	\$mod(int val, int divisor)	devolve o módulo (resto da divisão inteira) de 'val' por 'divisor'
Num	\$round(double valDecim, uint nrDig)	Arredonda um valor decimal a 'nrDig' de precisão.
Num	\$val(string exprNum)	Calcula o valor numérico da expressão enviada (pode conter operações matemáticas)

Funções String

str	\$left (string origem, uint nrDigitos)	Devolve os primeiros 'nrDigitos' de 'origem'
str	\$right(string origem, uint nrDigitos)	Devolve os últimos 'nrDigitos' de 'origem'
str	\$middle(string origem, uint start, uint nrDigitos)	Devolve a parte de 'origem' que começa em 'start', e não mais que 'nrDigitos' (1º caracter é numerado com a posição 1). Se start>tamanho de origem devolve uma string vazia
str	\$str(var const expr)	Concatena um conjunto de valores resultantes de uma expressão contendo variáveis, funções ou constantes alfa ou numéricos concatenados por '+'. Exemplo: \$str(1234+“.-+567+“ Lisboa” produz “1234-567 Lisboa”
str	\$case(uint nr, string array)	Devolve o elemento número 'nr' de 'array' (conjunto de strings separadas por ';'). Se nr=0 ou nr>nº de elementos da lista, devolve ""
bool	\$contains (string str, string amostra) \$\$contains(...)	Verifica se 'amostra' está contido em 'str'. A comparação será feita em modo case sensitive, ou case insensitive, se for usada \$\$contains. Esta função é idêntica a \$findStr, mas devolvendo um resultado booleano.
bool	\$contains (string str) \$\$contains(...)	Verifica se o valor da célula está contido em 'str'. e devolve '1' (true) se for verdade, ou '0' (false) no caso contrário.
num	find(string array, string amostra) \$\$find(...)	Procura, em modo case sensitive, a posição de 'amostra' em 'array' (conjunto de strings separadas por ';'), e devolve o nº do elemento do array onde encontrou, ou 0 se não foi encontrada. Para uma procura em modo case insensitive, usar \$\$find.
num	findStr(string origem, string amostra) \$\$findStr(...)	Procura, em modo case sensitive, se 'amostra' se encontra em 'origem', e devolve a posição de 'origem' onde a encontrou, ou 0 se não foi encontrada. Para uma procura em modo case insensitive, usar \$\$findStr.
num	\$findStr(string str) \$\$findStr(...)	Procura, em modo case sensitive, se o valor da célula se encontra em 'str', e devolve a posição de 'origem' onde a encontrou, ou 0 se não foi encontrada. Para uma procura em modo case insensitive, usar \$\$findStr.

Várias

undef	\$if(string exprLogica, undef resultTrue, undef resultFalse)	Avalia exprLogica, e selecciona se o resultado for 'true' devolve 'resultTrue', caso contrário, devolve resultFalse.
-	DefVar(string varName, string varValue)	Cria uma variável de nome 'varName' e atribui-lhe 'varValue'. Passa a disponibilizar o seu valor através de %varName

Apêndice B - Lista de variáveis globais ArtSOFT

Nome	Tipo	Descrição
Licence	Num	Nº de licença ArtSOFT.
WorkDate	Date	Data de trabalho, que poderá ser diferente da data de sistema.
SkuLen	Num	Tamanho do código de artigo, entre 2 e 18
OpcLen	Num	Tamanho do código opcional, entre 2 e 18
SkuName	AlfaN	Nome do código de artigo, ex. ‘referência’
OpcName	AlfaN	Nome do código opcional, ex: ‘código de barras’
SkuIni	AlfaN	Limite lógico inicial do ficheiro de artigos, por omissão ‘0’
SkuEnd	AlfaN	Limite lógico final do ficheiro de artigos, por omissão = ‘ZZZZZZ’
RoundPrc	Num	Nº de dígitos para arredondamento de preços de venda
RoundVal	Num	Nº de dígitos para arredondamento de valores
RoundQty	Num	Nº de dígitos para arredondamento de quantidades
CliLen	Num	Nº máximo de dígitos para o nº de cliente
ForLen	Num	Nº máximo de dígitos para o nº de fornecedor
FillLen	Num	Nº máximo de dígitos para o nº de filial
CliMaxNr	Num	Valor máximo que um nº de cliente pode conter, ex: 99999.
ForMaxNr	Num	Valor máximo que um nº de fornecedor pode conter, ex: 99999.
FilMaxNr	Num	Valor máximo que um nº de filial pode conter, ex: 999.
NrCliInd	Num	O nº de cliente atribuído aos clientes indiferenciados.
NrForInd	Num	O nº de fornecedor atribuído aos fornecedores indiferenciados.
CliName	Alfa	Nome dado à entidade ‘cliente’, como por exemplo: ‘Associado’
ForName	Alfa	Nome dado à entidade ‘fornecedor’, como por exemplo: ‘Fabricante’
VndName	Alfa	Nome dado à entidade ‘vendedor’, como por exemplo: ‘Consultor’
UserNum	Num	Nº de utilizador actual com login no ArtSOFT.
UserID	AlfaN	ID do utilizador actual com login no ArtSOFT.
UserName	Alfa	Nome do utilizador actual com login no ArtSOFT.
GrhTpFam	Array	Lista de tipos de familiares definidos na respectiva tabela, separados por ‘ ’
SkuMark	Array	Lista de marcas definidas para artigos, separadas por ‘ ’
CliMark	Array	Lista de marcas definidas para clientes, separadas por ‘ ’
ForMark	Array	Lista de marcas definidas para fornecedores, separadas por ‘ ’
TerMark	Array	Lista de marcas definidas para terceiros, separadas por ‘ ’
ImoMark	Array	Lista de marcas definidas para bens de imobilizado, separadas por ‘ ’
EmpMark	Array	Lista de marcas definidas para empregados, separadas por ‘ ’
IndMark	Array	Lista de marcas definidas para independentes, separadas por ‘ ’

Apêndice C - Lista de funções base ArtSOFT

Apêndice D – Lista de Funções

Uma função executa um determinado ‘trabalho’ pré-determinado, fornecendo um resultado em formato texto, ex: o nome de um artigo ou de uma conta, ou em formato numérico, ex: o preço de venda de um artigo, ou o saldo de uma conta. As funções assinaladas que retornam ‘*’ devolvem um texto com um conteúdo indeterminado, exemplo: uma imagem em Base64.

A lista de todas as funções disponibilizadas pelo ArtSOFT⁷ pode ser obtida a partir do serviço `ArtDB/_fnList`. Esta lista, tal como a lista de serviços disponibilizados depende da configuração do ArtXMLServer, na secção ‘[XMLSERVER]’, chave ‘ListaDLL =x;y;z’

As funções não podem ser chamadas directamente pelo XMServices, mas podem ser chamadas dentro de um query. No exemplo ao lado, pretende-se obter o stock disponível no armazém 1, dos artigos ‘x’, ‘y’, e ‘z’.

```
<TBL>
<item form='${$SkuValue(x,Avail,1) }' />
<item form='${$SkuValue(y,Avail,1) }' />
<item form='${$SkuValue(z,Avail,1) }' />
</TBL>
```

Se o resultado for do tipo numérico, a forma abreviada `form='função(parametros)'` é suficiente, pois assume por omissão `type='num'`. Já o pedido a uma função do tipo alfanumérico deve ser complementada com `type='text'`. Se isso não for feito, o query tentará converter o texto para número, e caso não consiga devolve ‘`ErroInvalidNum`’.

Descrição dos nós e atributos de output de todos os tipos de extractos:

Informação adicional sobre algumas funções e seus parâmetros

- **`$Valor(Activo, ...)`:** Esta função devolve o valor do conjunto de contas configuradas no elemento contabilístico de nome ‘Activo’ (neste caso).
- **`$SkuName(SKU,NameID,SkuType)`:** Devolve o nome ‘NameID’ de um artigo com o código ‘SKU’. Se o código é o da empresa, não é necessário indicar nada, ou em alternativa ‘Def’ (default), ou ‘Opc’ (opcional). NameID varia de 0=Nome principal a ‘4’-Nome opcional 4.
- **`$SkuPrice(SKU,PriceType,Wharehouse,SkuType)`:** Devolve um dos preços de PriceType, do armazém ‘x’. Lista de preços disponíveis⁸:
 - [0..9] - Preço de venda 0 .. 9. Se à data do pedido o preço for inválido (ter datas de inicio e fim e a data do pedido estar fora desse intervalo) o preço devolvido será 0.
 - LastCost - último preço de custo à entrada de armazém, na data do último recálculo de preços médios;
 - LastCNow - último preço de custo à entrada de armazém, na data do pedido;
 - LCostExW - último preço de custo à saída do fornecedor, na data do último recálculo de preços médios;
 - LCExWNow - último preço de custo à saída do fornecedor, na data do pedido;
 - MeanCost - preço médio de custo ponderado, na data do último recálculo de preços médios;
 - MeanCNow - preço médio de custo ponderado, na data do pedido;
- **`$SkuValue(SKU,ValueType,Wharehouse,IniMonth,EndMonth,SkuType)`:** Devolve os valores acumulados do artigo ‘SKU’, do tipo ‘ValueType’, para a lista de armazéns ‘Wharehouse’, entre os meses ‘IniMonth’ e ‘EndMonth’. A lista de armazéns poderá conter apenas ‘0’ para os acumulados de toda a empresa, um número de armazém, ex: ‘1’, para os acumulados apenas desse armazém, ou uma lista de intrevalos de armazéns, ex: 1:5;9:10:30, em que se pretendem os valores dos armazéns 1 a 5, do armazém 9, e dos armazéns 10 a 30. Lista de tipos de valor disponíveis:
 - QtEnt - Quantidades entradas;
 - QtOut - Quantidades saídas;
 - QtVnd - Quantidades vendidas;
 - Stock - Stock total em armazém;
 - Avail - Stock disponível em armazém (Stock total em armazém - Quantidades já cativadas);
 - VLEnt - Valores Entrados;
 - VLOut - Valores Saídos;
 - VLVnd - Valores Vendidos;
 - Profit - Lucro Bruto.

⁷ Poderão ser acrescentadas as que forem necessárias através de um serviço plugin.

⁸ Nota: Os dados de LastCNow, LCExWNow e MeanCNow são de baixa qualidade uma vez que poderão ser actualizados por qualquer correção, e os restantes só terão alguma qualidade se for feito um encerramento da data da gestão comercial na data em que foram feitos os recálculos de preços médios, evitando assim que antes dessa data ainda haja alteração de valores.

Apêndice E - Formulários HTML

A ‘impressão’ de um documento deverá ser feita com o formato que o utilizador pretender, e ainda, poder ter várias formas, de acordo com o fim a que se destina (factura, talão, etc). Para conseguir esse objectivo, optou-se por efectuar a transformação de um ‘formulário template’ tendo como suporte um ficheiro HTML, que conterá todo o design do documento, e indicações de como essa transformação ocorrerá.

Para colocar essas indicações, será usada a propriedade HTML ‘title’ que serve para comentar determinada zona ou sugerir dicas quando se coloca o ponteiro do rato sobre ela, muito útil quando este é usado para formatação Web, mas irrelevante na apresentação ou impressão de formulário de documentos.

Assim, o atributo ‘title’ será usado para definir uma fórmula ou variável que irá ser usada na próxima zona delimitada por ‘<...>’. Exemplo: <td title="DocFch.Tot.Doc"><999,999.99</td>. Neste exemplo, seria procurada a variável ‘DocFch.Tot.Doc’, e o seu valor colocado no lugar onde está agora um protótipo do valor ‘<999,999.99>’, e removida a propriedade ‘title’, e, supondo que a variável tinha o valor de 123,456.50, resultaria em: <td>123,345.50</td>.

A utilização da propriedade ‘title’ tem uma vantagem adicional para manutenção do formulário: o facto de esta apresentar dicas quando o rato está sobre a sua zona de influência, irá ser muito útil quando o formulário for apresentado num browser, pois irá mostrar a fórmula ou variável associadas a cada zona.

Para que essa transformação possa ocorrer, consideremos que o documento será dividido em três partes: 1-um **cabeçalho**, que conterá tudo o que esteja antes da linha de movimentos; 2-A(s) **linha(s)** de movimentos; 3-Fim do documento, que conterá tudo o que esteja após a linha de movimentos.

O cabeçalho e o fim são transformados uma única vez. A linha de movimentos é transformada tantas vezes quantas as linhas presentes no documento.

Torna-se assim necessário definir onde começa a linha de movimentos (uma ‘<tr>’ ou ‘table row’). Para isso, marca-se a linha de movimentos com a indicação: <tr title="[DocLan]*x">, em que a parte ‘*x’ só é necessária se forem necessárias ‘x’ linhas HTML para cada linha de movimentos (por exemplo, em formulários tipo ‘talão’, ver o Doc. 3, na pág. 26). Obviamente, existindo a indicação que são usadas ‘x’ linhas, deverão existir definidas de seguida ‘x’ linhas (<tr> ... </tr>).

É também possível definir um ‘setup’ do documento, e um conjunto de constantes de formatação, na tag <body>:

```
<body title="setup='LanObs=S,Lingua=P,TextTab=1' dec='.' V01='##,###.##' V02='###' V03='##,###.##-' ">
```

1. Setup: permite definir três parâmetros importantes:

- LanObs='[N]|S': Indica se vai ser usada uma coluna específica para observações do lançamento. Se não for, o texto de observações irá ser concatenado ao da descrição do produto.
- Lingua='B|C|E|F|P|[p]' (B=Francês/Belga, C=Castelhano, E=Inglês(English), F=Françês, P=Português, p=Português ligeiro): Permite definir a língua em que o extenso do documento deve ser produzido (se for necessário extenso).
- TextTab='xx': Indica o nº da tabela de personalização do formulário existente no ArtSOFT em ‘Configurações/Tabelas empresa/Listagens/Personalização de formulários’. Só será util se o formulário utilizar as variáveis ‘IdEmp.Linha00 .. IdEmp.Linha19’, permitindo ao utilizador alterar facilmente o texto de identificação da empresa.

Dec: Permite definir qual o carácter a ser usado para separador decimal, por omissão é ‘.’, mas pode ser também ‘,’ e esta definição irá ser importante para interpretar as máscaras de formatação, do tipo ‘##,##.##’.

V01, V02, ...Vxx: Variáveis que contém máscaras de formatação. A formatação de quantidades, taxas de IVA e valores irá ser usada em todo o documento, e as formulas irão contê-las, ou como constantes, ex: “DocFch.Tot.Doc@##,##.##” ou “DocFch.Tot.Doc@V01”. Torna-se evidente que, se for necessário alterar a formatação em todo o documento, será muito mais fácil fazê-lo apenas num local.

Variáveis e Fórmulas

Uma variável ou campo é simplesmente o nome completo de uma variável, tal como esta é usada nos queries, serviços, gerador de relatórios, etc⁹. ‘title’ pode referir uma variável, mas também pode ser usada uma fórmula algébrica com outras variáveis ou constantes:

- a) title=(DocFch.Tot.Bruto-DocFch.Tot.DescP)/1000@###,###
- b) title=DocLan.Div.Descric + ' ' + Qtd.Med0 + 'x' + Qtd.Med1 + 'x' + Qtd.Med2 + Div.Obs

No caso a) subtotal com o total líquido menos o total de descontos de produto expresso em milhares. No caso b) pretende-se concatenar a descrição com as três medidas e juntar-lhe as observações do lançamento. Nota: nas formulas de concatenação não faz sentido o uso de parêntesis para definirem regras de prioridade. Como aqui é necessário o uso de plicas para delimitar texto constante, o conteúdo de ‘title=’ tem que ser delimitado por aspas.

Formatação de campos

Numéricos

Formato	Valor	Resultado ¹⁰
###,###	9.9 987654.32 -6666.778	10 987,654 -6,667
###,###.##	9.9 987654.32 -6666.778	9.99 987,654.32 -6,666.78
###,###.##-	9.9 987654.32 -6666.778	9.99 987,654.32 6,666.78-

Formato	Valor	Resultado
(###,###.##)	9.9 987654.32 -6666.778	9.99 987,654.32 (6,666.78)
(*###,###.##)	9.9 987654.32 -6666.778	*****9.99 *987,654.32 (***6,666.78)
(=###,###.##)	9.9 987654.32 -6666.778	=====9.99 =987,654.32 (==6,666.78)

Alfanuméricos:

A formatação alfanumérica não tem qualquer sentido usada para ajustar textos à direita ou ao centro, pois em HTML, os atributos de ‘align’: ‘left’, ‘center’ e ‘right’ fazem isso de forma mais eficiente. No entanto, poderá ser necessário alinhar determinado texto à esquerda, ao centro ou à direita, e preencher determinado espaço com um determinado carácter. Para isso incluiu-se um formato específico para esse efeito:

“tamanho,alinhamento,preenchimento”, em que:

1. tamanho: o tamanho em caracteres do resultado pretendido;
2. alinhamento: ‘L’=Left/Esquerda, ‘C’=Centrado, ‘R’=Right/Direita
3. carácter de preenchimento. Se este carácter for ‘@’, deve usar-se ‘@@’

O formato “64,C,=” significa centrar uma string em 64 caracteres, cortando-a se for maior, ou preenche-la com ‘=’ à esquerda e/ou à direita, até perfazer o tamanho indicado.

Exemplos:

Formato	Texto original	Resultado
32,L,=	Boas Festas	Boas Festas=====
32,C,=	Boas Festas	=====Boas Festas=====
32,R,=	Boas Festas	=====Boas Festas=====
32,L C R,=	Texto muito extenso que ultrapassa 32 caracteres	Texto muito extenso que ultrapas

Datas

A formatação de datas é igual à descrita na pág. 50 em ‘Funções internas de XMLReports / Formatos de Datas’, coluna ‘fmtDatas’, ou no formato reduzido, coluna ‘fmtDatas (alt)’.

⁹ Nos cabeçalhos das exportações ASCII só é usado o nome do campo e não o nome completo da variável (Tabela.Campo), porque sendo a exportação / importação de uma determinada tabela, o contexto (namespace) de todas as variáveis já é conhecido: é o da própria tabela.

¹⁰ Apesar de as máscaras poderem definir espaços à esquerda de um número para o alinharem, este método não foi usado, devendo usar-se o atributo align="right" para esse efeito, porque produz resultados mais estáveis em todos os browsers e impressoras, para além de produzir ficheiros .HTML mais pequenos.

Exemplo de formulário HTML

```

<html>
<head>
  <title>...</title>
  <style type="text/css"> ... </style>
</head>
<body title="setup='LanObs=S,Lingua=P,TextTab=1' dec='.' V01='##,###.##' V02='###' V03='##,##.##-.' ">
<table cellspacing="0" cellpadding="0" width="725" border="0">
<tr>
  <td width="232" align="center" rowspan="2">
    <p title="IdEmp.Linha00"><font size="1"><b>«Empresa0»</b></font></p>
    <p title="IdEmp.Linha01"><font size="1"><b>«Empresa1»</b></font></p>
    <p title="IdEmp.Linha02"><font size="1"><b>«Empresa2»</b></font></p>
    <p title="IdEmp.Linha03"><font size="1"><b>«Empresa3»</b></font></p>
  </td>
</tr>
...
<tr>
  <td title="[DocLan]*x">
    <td title="Lst.CodUso" width="103" align="right"><p align="left"><b>«Codigo»</b></p></td>
    <td title="DocLan.Qtd.Real@V01" width="48" align="right"><font size="1"><b>«Quant»</b></font></td>
    <td title="DocLan.IVA.Taxa@V02" width="35" align="right"><font size="1"><b>«ivax»</b></font></td>
    <td title="DocLan.Val.TtLiq@V03" width="57" align="right"><font size="1"><b>«TtIliquido»</b></font></td>
  </tr>
...
<tr>
  <td align="center" width="50">Totais</td>
  <td title="DocFch.IVA.Inc@V01" align="right" width="149"><b>«Incid. tot.»</b></td>
  <td title="DocFch.IVA.Val@V01" align="right" width="138"><b>«Valor tot. IVA»</b></td>
  <td align="center" width="240" colspan="2"><b>TOTAL DO DOCUMENTO</b></td>
  <td title="DocFch.Tot.Doc@V03" align="right" width="133"><i><b>«TotalDocum.»</b></i></td>
</tr>
</table>
</body>
</html>

```

Apêndice F - Informação adicional sobre chaves de tabelas ArtSOFT

Indicada uma tabela válida, o serviço *ArtDB/_tblDesc* (pág. 8), informa quais as chaves (indexes) e campos (colunas) da mesma.

Alguns tipos de segmentos de chave não ficam suficientemente explícitos na informação sucinta dada pelo serviço. Neste caso, é necessário descrever estes aqui.

Lançamentos de Contabilidade (CtaLan)

Lançamentos por Diários/Documentos	ExtDiaDoc	CtaLan ExtDiaDoc Mes=2:8 Diario=1:20 TpDoc=A1:A999 Data=20010101:20091231 SoCC=T
Lançamentos por Tipos de documento	TipoDoc	CtaLan TipoDoc TpDoc=A1:A999 NrDoc=1:999999 SoCC=T
Datas de Vencimento	DataVenc	CtaLan DataVenc AICTa=1234 TpDoc=A1:A999 NrDoc=101:991231 Tipo=T
Lançamentos pendentes de ligações	PendLink	CtaLan PendLink Link=xxxxxxx AICTa=1234 Data=19990101:20991231
Ligações entre lançamentos	Links	CtaLan Links Link=xxxxxxx NrLanc=1234

SoCC ¹¹	1=Em lançamentos de C/C, apenas o lançamento de ordem 0	0=Todos os lançamentos do documento (custos, contrapartidas, etc.)
Tipo	T=Todos os lançamentos, incluindo os já regularizados	(por omissão)=Só lançamentos ainda não regularizados

xxxxxx	Significado
Deposito	Deposito
Letra	Letra
Reforma	Reforma
ReavActivo	Reavaliações de Activos
LetraDevol	Devolução de Letra
LetraPagFact	Letra/Pagamento de Factura
LetraDesconto	Desconto de Letra

xxxxxx	Significado
LetraPagam	Pagamento de Letra
LetraNPaga	Letra Não Paga
LetraDevol	Devolução de Letra
LetraReforma1	Letra/Reforma1
LetraReforma2	Letra/Reforma2
LetraRecambio	Recambio de Letra
LetraSubstit	Substituição de Letra

¹¹ Para especificar um valor lógico, pode usar-se a forma booleana: 0 ou 1, mas também: F=False, N=No/Não em vez de '0', ou T=True, V=Verdadeiro, Y=Yes, S=Sim em vez de '1'.

Apêndice G - Autenticação no servidor

Modo produção

Para manter um nível de segurança adequado, o servidor XML ArtSOFT necessita de reconhecer quem está a efectuar os pedidos, afim de manter ‘fora’ do sistema eventuais hackers capazes de provocarem algumas dores de cabeça ao administrador de sistema.

Para isso, foram criados os seguintes algoritmos:

- Validação da lista de IP's autorizados a comunicar (opcional): para isso deve criar-se uma chave ‘HostsList’ no ficheiro de configuração. Esta deve conter a lista de IP's autorizados a efectuarem pedidos, separados pos ‘;’. Exemplo:
 - HostsList=localhost;76.5.43.210;192.168.1.2;www.artsoft.pt
- Validação da lista de ‘utilizadores’ autorizados a comunicar com o servidor (opcional): Todas as operações efectuadas por estes utilizadores irão aparecer no ArtSOFT como efectuadas pelo utilizador ‘0’, ou seja, o sistema. Para isso é necessário criar a chave ‘UsersList’ no ficheiro de configuração com a lista de utilizadores e respectivas palavras-chave encriptadas, na forma ‘User1:Psw1; ... UserN:PswN’. Exemplo:
 - UsersList=Joaquim: 4fH6kC4Dim2V2UBfw; Manuel:C4Dim2V2UBfw1SrASj
- É possível juntar à lista de utilizadores válidos a lista de utilizadores normais ArtSOFT, desde que tenha accionada a respectiva permissão. Todas as operações irão aparecer no ArtSOFT como tendo sido feitas pelo utilizador correcto. Para isso, basta adicionar a chave ‘UseArtUsr=S’ no ficheiro de configuração. NOTA 1: para evitar uma enorme confusão, não deverão existir nomes duplicados. Se isso acontecer, serão usados em primeiro lugar os nomes constantes da ‘UsersList’. NOTA 2: a versão do servidor multiempresa não suporta a lista de utilizadores ArtSOFT.

É de notar que ‘HostsList’ pode conter o IP em formato numérico, ou o nome do host, mas esta é lida apenas quando o serviço é inicializado, mantendo-se estática durante o funcionamento. Por essa razão, não devem ser incluídos na lista nomes de hosts com IP's dinâmicos, pois pode ocorrer uma mudança de IP, que não constará da lista.

Quando um cliente se conecta ao servidor, deve enviar-lhe um HTTP GET para o recurso ‘login’, e no cabeçalho, deve ser adicionada uma chave ‘Name’ com o nome do utilizador que se pretende ligar. Adicionalmente, pode juntar-se também uma chave ‘XmlIndent’ com o valor da identação que os nós (nodes) dos streams XML devem ter, que por omissão, é -1 (sem CR/LF e sem identação de nós), podendo ser enviados valores razoáveis de ‘0=Só com CR/LF’, ‘1...3=CR/LF seguido de 1..3 espaços por cada nível.

Em resposta, o servidor envia um status ‘200 OK’, uma chave no cabeçalho de nome ‘Digest’, contendo um desafio com 20 a 22 dígitos. Seguidamente, o cliente deverá colocar no cabeçalho da sua primeira mensagem uma chave de nome ‘Digest’. O valor para esta chave é formado por 16 ou 20 bytes codificados em hexadecimal (32 / 40 dígitos), e é calculado, consoante o algoritmo escolhido, segundo o seguinte pseudocódigo:

MD5: efectuar o ‘digest’ com MD5 de “UserName”, “Password”, “Desafio” para as posições 0, 16 e 32 de um ‘byte array’ com 48 bytes de tamanho, efectuar de novo o ‘digest’ desse buffer, e converter o resultado para hexadecimal:

```
char buffer[16*3], output[16], result[32+1]
MD5 (in UserName, out &buffer[00])
MD5 (in Password, out &buffer[16]) *
MD5 (in Desafio , out &buffer[32])
MD5 (in buffer , out output)
Hexa(in output , out result)
```

SHA1: efectuar o ‘digest’ com SHA1 de “UserName”, “Password”, “Desafio” para as posições 0, 20 e 40 de um ‘byte array’ com 60 bytes de tamanho, efectuar de novo o ‘digest’ desse buffer, e converter o resultado para hexadecimal:

```
char buffer[20*3], output[20], result[40+1]
SHA1(in UserName, out &buffer[00])
SHA1(in Password, out &buffer[20]) *
SHA1(in Desafio , out &buffer[40])
```

```
SHA1(in buffer , out output)
Hexa(in output , out result)
```

Se for pretendido enviar vários comandos de seguida, deve colocar-se no cabeçalho ‘Keep-Alive=xx’, em que xx é o número de segundos que a ligação se deve manter (máx: 30 segundos). No último pedido o cabeçalho não deve conter este atributo, indicando que o servidor após processar e responder a esse comando pode encerrar ‘graciosamente’ a ligação.

Nota: ‘Name’ só pode ser enviado na 1^a mensagem (a da ligação), e ‘Digest’ só pode ser enviado na 2^a mensagem, e apenas uma vez, senão, a ligação será encerrada!

Modo debug

No modo debug, o serviço não executa nenhuma função, mas permite testar os algoritmos de login e digest, afim de verificar o seu bom funcionamento. Para isso, deve enviar-se o pedido para ‘logdebug’ em vez de ‘login’, com um user ‘ART’ e password ‘SOFT’. Neste modo, para que os valores possam ser comparáveis, o servidor envia um desafio sempre constante “ABCDEFGHIJKLMNOPQRSTUVWXYZ”. O resultado obtido deverá ser:

- MD5: 7e cc a8 a4 00 07 9c 6a 31 88 44 3d 08 e3 6e 73
- SHA1: 3d f2 5f 88 79 0d aa ea 81 f9 68 cd c1 a2 1f 15 65 df 64 32

Teste de comunicação

As aplicações utilizadoras poderão ter uma funcionalidade muito simptática para quem está a configurar o sistema, colocando um botão ou opção de menu ‘Testar conexão’. Para informar o utilizador qual o estado da ligação, pode fazer-se o login em modo debug, e se a resposta for negativa, existe um problema com o endereço do host ou da porta em que o serviço está instalado. Se a resposta for positiva, já é possível contactar o serviço, mas ainda não se sabe se o ‘UserID’ e ‘Password’ estão correctos. Para isso, pode fazer-se nova ligação para ‘logtest’ com o utilizador e password correctos. Neste modo, o serviço também não executa nenhuma função, mas, por motivos de segurança, todos os pedidos de ‘logtest’ são enviados para o ficheiro de log e notificações.

Algoritmos de validação de clientes

O servidor permite a utilização dos algoritmos MD5 e SHA1 para validação de clientes, devendo ser escolhido o SHA1, existindo o MD5 para suporte a aplicações e plataformas de desenvolvimento mais antigas.

Um algoritmo de hash só tem uma via, ou seja, dado um texto ou documento, é sempre possível gerar um hash para o mesmo, mas, dado um determinado hash não é possível¹² determinar qual o texto que lhe deu origem.

Para ajudar nessa escolha, é necessário ter em atenção ao seguinte:

- Se tiver acesso ao SHA1 na sua plataforma de desenvolvimento, use-o, porque é mais seguro que o MD5.
- No pseudocódigo, o passo assinalado com “*” pode ser substituído por uma cópia do hash da password previamente calculado, evitando que no código, ou na sua configuração, exista a password em pleno texto (O utilitário ‘XMLwsClient.exe’ no botão ‘outros logins’, contém uma secção para o cálculo de hashes).
- Mesmo com a password ‘digerida’, é muito importante escolher uma password forte (letras maiusculas/minusculas/numeros e simbolos), para evitar ataques do tipo ‘dicionário’¹³

¹² Não deveria ser possível, mas como se irá mostrar de seguida, para pequenos textos como as passwords, infelizmente existem formas de o fazer...

¹³ Percorre-se uma lista de números ou palavras habitualmente usadas, faz-se hash de cada elemento e compara-se com o hash da digestão. Se for igual, a password foi localizada...

Apêndice H – Servidor multiempresa

O serviço ArtExec cria um ambiente de trabalho para uma empresa, e por consequência, o servidor ***ArtExecServerMin*** só disponibiliza serviços XML sobre a base de dados dessa empresa.

Em alguns casos especiais, é necessário trabalhar em bases de dados de várias empresas (caso dos grupos de empresas, SGPS e empresas prestadoras de serviços de gestão, gabinetes de contabilidade, etc.).

Para isso foi criado o servidor ***ArtExecServerME***, com funcionalidades exactamente iguais ao anterior, mas com uma particularidade: quando se efectua o login, pode especificar-se qual a empresa a que se deve conectar. No entanto, veremos adiante porquê, este servidor necessita de ser o único que o ArtExec esteja a executar.

Este servidor impõe duas regras diferentes do ***ArtExecServerMin***: O nome do utilizador irá ser o nome da Base de Dados da empresa, precedido de ‘~’, e deverá existir um novo atributo na mensagem de login, com WorkYear=xxxx, em que xxxx representa o ano pretendido, mas não deverá ser superior ao ano actual nem inferior ao ano actual menos cinco. Ex: estando em 2030, só permite trabalhar na base de dados destino nos anos 2029 e 2030.

Quando este servidor recebe uma ligação e autentica um cliente, extrai o ‘~’ inicial tenta conectar-se à base de dados destino identificada no ‘username’ restante, e ano fornecido pelo atributo ‘WorkYear’. Nessa altura, se existisse qualquer serviço a ser executado, poderia correr o risco de, em vez de efectuar o seu trabalho sobre a base de dados base, o efectuasse sobre a base de dados que tivesse sido seleccionada por uma ligação ao serviço ***ArtExecServerME***, com consequências difíceis de prever, mas cuja probabilidade de acontecer não é zero.

Assim, este serviço ao ser carregado, vai ser egoista e verificar se é o ‘único filho’ a ser executado por aquela instância do ArtExec, e se não for verdade, falha o arranque, assinalando o facto no ficheiro de log. No entanto, se os ‘irmãos’ forem carregados depois deste (ou por estar à frente deste no ficheiro de configuração, ou ter sido carregado dinamicamente via browser), ele já não os pode detectar, e neste caso o risco continuaria a existir. Assim, quando chegar qualquer pedido de login, este volta a verificar se existem serviços em concorrência com este, e se isso acontecer, este servidor pára de imediato, assinalando ao cliente o erro ‘500-Internal Server Error’.

Nota: O servidor ‘***ArtExecServerMin***’ não irá permitir logins de utilizadores não começem com ‘~’, evitando assim que por erro um cliente se ligue ao servidor errado e efectue operações eventualmente desastrosas na base de dados errada. Por sua vez, o ‘***ArtExecServerME***’ não permite logins com nomes que não começem com ‘~’.

Apêndice I – Comunicação Segura

Para manter um nível de confidencialidade adequado, o servidor XML ArtSOFT permite que todos os dados (o XML transportado no ‘body’ da mensagem HTTP) sejam encriptados.

Pode ser configurado para que todas as mensagens devam ser encriptadas (com Encript=S na configuração), ou a pedido do cliente (enviando no tipo de mensagem ‘bin/xml’ em vez de ‘text/xml’).

Independentemente da configuração, determinados serviços só podem ser executados mediante uma comunicação segura, pois são demasiado perigosos para serem expostos a escutas na rede, como os serviços de atribuição de palavras-passe e todos os relacionados com recursos humanos.

Para efectuar as operações de cifra e descifra, é necessária uma chave com 20 bytes / 160 bits. Para a construir, deve fazer-se:

- SHA1: criar um array temporário de 40 bytes, e um array para a chave com 20 bytes. Colocar nos primeiros 20 bytes do array temporário o hash SHA1 da password do cliente¹⁴, (dado constante conhecido de ambos os lados, mas que nunca transitou na rede), nos segundos 20 bytes o digest do desafio enviado pelo servidor (dado aleatório variável por cada ligação), e por fim colocar no array chave o digest SHA1 do array temporário.
- MD5: criar um array temporário de 32 bytes, e um array para a chave com 20 bytes. Colocar nos primeiros 16 bytes do array temporário o hash MD5 da password do cliente, (dado constante conhecido de ambos os lados, mas que nunca transitou na rede), nos segundos 16 bytes o digest do desafio enviado pelo servidor (dado aleatório variável por cada ligação), e por fim colocar nos primeiros 16 bytes do array chave o digest MD5 do array temporário, e complementar os 4 bytes restantes com a cópia do 4º, 8, 12º e 16º byte do mesmo array.

Para cifrar a mensagem do body deve efectuar-se a operação XOR de cada byte da string XML com o respectivo byte da chave. Quando esgotada a chave, deve voltar-se ao inicio desta. O resultado da encriptação deve ser enviado na mensagem HTTP com o ‘Content-Type=bin/xml’.

Para descifrar a mensagem do body, depois de recebido o stream XML cifrado, deve efectuar-se o mesmo procedimento descrito para a cifra (efectuando XOR de cada byte do stream com o respectivo byte da chave). Depois desta operação, obtem-se o XML original.

A mensagem vai passar a circular em binário, e qualquer erro de cifra ou adulteração da mensagem vai produzir um resultado imprevisível. Para evitar estar a fazer parsing de um XML inválido, é adicionada uma chave no cabeçalho HTTP com o checksum simples de 32 bits (soma de todos os bytes da mensagem antes de cifrada módulo 0xFFFFFFFF). Essa chave deve chamar-se ‘Checksum’ e conter o valor do checksum em hexadecimal, ex: “checksum: 89abcdef”.

Assim, o cliente, ao receber uma mensagem cifrada deve descifrá-la, efectuar o checksum do resultado e verificar-lo com o obtido no cabeçalho da mensagem. Se estiver não estiver correcto, deve fechar a ligação e mostrar um erro e / ou gerar um log, consoante as circunstâncias. Quando enviar uma mensagem, deve calcular o checksum da mensagem, cifrá-la, juntar a chave ‘Checksum’ e o respectivo valor e por fim enviar a mensagem.

¹⁴ NOTA: Sendo a password um dado constante, esta operação pode já estar previamente feita, e assim a aplicação cliente apenas coloca os bytes pré-digeridos nos primeiros 20 bytes do array temporário, evitando assim ter a password exposta em pleno texto na aplicação (e portanto vulnerável a ataques) ou evitando o trabalho de a manter encriptada na aplicação, que pode também estar vulnerável se o algoritmo de cifra for fraco. Mais sobre este assunto em ‘Algoritmos de validação de clientes’, na pág. 45.

Apêndice J - Configuração do servidor

No ficheiro de configuração do serviço *ArtExec*, deve ser acrescentada a seguinte secção:

```
[XMLSERVER]
Notify      =1,2,3
NotifMail=1,2,3
NotifSMS =1,2,3
XMLPort    =2600
TimeOut     =30
XmlIndent=-1
ServerNm   =www.artsoft.pt
Digest      =MD5
Encoding    =UTF-8
Encrypt     =S
ListadLL  =ArtDB_D,ArtDB_S
LogFile    =x:\path\logFile.txt
HostsList=localhost;10.1.1.250;www.artsoft.pt
UsersList=User: PasswordCifrada;Outro:PasswordCifrada
```

- **Notify, NotifyMail, NotifySMS:** Lista de até cinco utilizadores a receberem mensagens de alerta ou erro durante o funcionamento do serviço.
- **XMLPort:** Porta TCP que o servidor está à espera de ligações.
- **TimeOut:** Tempo máximo de espera por um pedido. Não pode exceder 30, senão, assume 30 segundos.
- **XMLIndent:** A indentação que o stream XML de resposta deve conter. -1=Sem CR/LF, 0=Com CR/LF, 0 espaços, >0=Com CR/LF + x espaços. Este parâmetro pode ser definido por cada sessão, se o cliente enviar no cabeçalho da mensagem “`Xmllnident=xx`”.
- **ServerNm:** O nome com que o servidor se identifica ao cliente, no cabeçalho da mensagem com o nome ‘`Host=xxx`’. Este campo é obrigatório em HTTP 1.1.
- **Digest:** escolher um: **SHA1, MD5** (mais informações sobre estes algoritmos, ver acima).
- **Encoding:** por omissão ‘`ISO8859-1`’. Se for pretendido “`utf-8`”, colocar esse valor neste parâmetro. Nota: em modo ‘`utf-8`’, o servidor passa a converter todos os pedidos para ‘`ISO8859-1`’, efectuar o processamento e voltar a codificar a resposta em ‘`utf-8`’, logo só deve mesmo ser usado, se o sistema cliente trabalhar totalmente neste formato (se possuir dados em ‘`ISO8859-1`’ já tem que efectuar conversões, pelo que também pode converter as respostas).
- **Encrypt:** Indica que toda a comunicação para este servidor deve ser encriptada. Sobre comunicação segura, consultar o Apêndice I - Comunicação Segura.
- **ListaDLL:** A lista de DLLs, separados por ‘,’ ou por ‘;’ , necessários aos serviços que se pretende disponibilizar. Abaixo é apresentada a lista de DLL’s, os seus respectivos serviços e dependências.
- **LogFile:** O nome do ficheiro de log / auditoria. Se este parâmetro for indicado, o servidor guarda neste ficheiro exactamente todos os pedidos recebidos e todas as respostas enviadas, sendo um precioso auxiliar para quem implementa um serviço tirar quaisquer dúvidas¹⁵ sobre o que o servidor está a receber / enviar. Só deve ser usado em fase de implementação, senão, atingirá rapidamente tamanhos muito grandes.
- **HostsList:** O nome ou endereço IP dos computadores autorizados a comunicar com o servidor. Se este campo for vazio ou inexistente, assume-se que são todos.
- **UsersList:** Lista de utilizadores e respectivas passwords cifradas, no formato ‘`user1:pswCifrada1; user2:pswCifrada2; user3:pswCifrada3`’. Para gerar a cifra da password, usar o utilitário **TextCrypt.Exe**. No caso de se estar a utilizar o servidor multi-empresa o nome de utilizador, passa a ser o nome da base de dados precedido de ‘~’, ficando na forma `-NomeBD:pswCifradaEmpresa`.
- **UseArtUsr:** Permite juntar à lista de utilizadores todos os utilizadores registados do ArtSOFT e que contenham a respectiva permissão ligada. Para mais detalhes sobre esta opção, consultar ‘Apêndice G - Autenticação no servidor’.

Se forem usados serviços que processem formulários, deverá existir na secção [CONFIG]:

```
[CONFIG]
PathFrm=\Artsoft\Forms
```

Lista de DLL's, serviços e dependências

- **ArtDB_A** - Serviços relacionados com a gestão de artigos e stocks. Depende de **ArtDB_B**.
- **ArtDB_B** - Queries. Serviços informativos. Serviços relacionados com tabelas de configurações.
- **ArtDB_T** - Serviços relacionados com a gestão de terceiros. Depende de **ArtDB_B**.
- **ArtDB_P** - Serviços relacionados com a contabilidade. Depende de **ArtDB_B** e **T**
- **ArtDB_D** - Serviços relacionados com documentos. Depende de **ArtDB_A, B, T e P**
- **ArtDB_S** - Serviços relacionados com salários. Depende de **ArtDB_T e P**

¹⁵ O que o servidor recebe em muitos casos não é exactamente aquilo que o cliente enviou, porque poderão existir programas pelo caminho que alterem a mensagem silenciosamente. Por exemplo, um cliente envia um pedido através de um servidor WEB que está configurado para proteger ‘SQL Injection’, colocando antes de cada aspa um ‘\’ (backslash), que irá invalidar toda a sintaxe XML recebida pelo servidor, que passará a rejeitar todos os pedidos sem razão aparente...

- ArtDB_V - Serviços relacionados com vendedores. Depende de ArtDB_T e S
- DocPrintCmd - Impressão interactiva de documentos de facturação. Depende de ArtDB_A, B, T, P e D.
- DocPrintSvc - Serviço de impressão de documentos de facturação. Depende de ArtDB_A, B, T, P e D.
- SqlQrySvc - Serviço de queries SQL.

Apêndice K – Validação de Logins

Afim de manter a confidencialidade da password de um contacto de terceiro, utilizador ArtSOFT, empregado ou independente entre um servidor WEB e o browser, sem ter que recorrer à encriptação da comunicação via SSL/TLS (evitando o uso de certificados digitais¹⁶), este serviço permite que a plataforma de internet da empresa delegue nele a tarefa de acreditação do utilizador, tornando muito fácil a manutenção de utilizadores a partir do próprio ERP, mantendo estes associados à ficha de terceiro.

Assim, a plataforma WEB deve apresentar algures uma entrada para a área reservada, com pelo menos duas entradas: uma para a identificação do utilizador, e outra para a password, e o correspondente botão ‘OK’ ou similar.

Ao enviar o formulário que contenha estes campos para o browser, deve incluir também um valor invisível, que deverá ser gerado nesse momento, constituindo este o ‘seed’. Para que tenha a segurança desejada, este deverá ter pelo menos 16 dígitos gerados aleatoriamente por ferramentas criptográficas. Deve ser também enviado ao browser o código¹⁷ que execute o algoritmo de ‘hash’ descrito a seguir.

O browser deve então pedir ao utilizador o identificador e a sua password, formatá-los da seguinte forma “[UserID]~[Password]”¹⁸, calcular o respectivo digest em SHA1 e guardá-lo nos primeiros 20 bytes do array ‘Result’. Se seguida calcular o digest SHA1 do ‘seed’ (desafio) e colocá-lo nos segundos 20 bytes, efectuando depois o digest SHA1 do array resultado sendo o resultado colocado no array buffer. Este resultado deve ser convertido para hexadecimal (40 dígitos) e juntamente com o UserID, deve ser enviado pelo browser ao servidor WEB. Este guardou em sessão o ‘seed’ que enviou ao browser, recebe o ‘UserID’ e o ‘digest’, e devolve estes três elementos ao servidor XML ArtSOFT. Pseudo-código:

```
char buffer[20],result[40+1]
SHA1(in [UserID]~[Passw], out &result[0])
SHA1(in Seed, out & result[20])
SHA1(in result, out buffer)
Hexa(in buffer, out result)
```

Nota 1: Depois de enviar o pedido ao servidor XML ArtSOFT, deve limpar de imediato o ‘seed’, e agir de acordo com a resposta recebida. Deve ter em atenção que a recepção de um POST de login sem que o servidor tenha um ‘seed’ válido, pode tratar-se de um ataque de reply (algum, entre o servidor WEB e o browser) guardou uma resposta válida e está a tentar obter um login ilegal.

Nota 2: O uso da técnica de não guardar o ‘seed’ em sessão e enviá-lo num campo input do tipo ‘hidden’ (mesmo que encriptado) para que o browser o devolva juntamente com os restantes atributos, fica sujeito um ataque de reply sem hipótese de ser detectado...

Nota 3: Estes pedidos de login do servidor Web ao XML Server devem ser encriptados para evitar que alguém na rede interna intercepte estes pedidos e ganhe acesso aos dados que estes protegem, que no caso de um ‘contacto de terceiro’ não é muito útil, mas em *Utilizadores ArtSOFT, Empregados e Independentes*, seria muito apetecível (e com graves consequências). Por essa razão, os serviços

¹⁶ Nota: Deve recorrer-se sempre à comunicação encriptada sempre que o tipo de informação transferida tenha carácter de confidencialidade e valor que o justifique. O processo aqui descrito é susceptível a ataques difíceis, mas possíveis, de um hacker construir uma réplica da página WEB a atacar, conseguir desviar o utilizador para este site, e capturar-lhe o login. Porém este tipo de ataque é sempre possível mesmo que a página use SSL/TLS, pois o utilizador médio nunca presta atenção se a página tem ou não um ‘cadeado’.

¹⁷ Em javascript por exemplo.

¹⁸ A formatação é feita desta forma para evitar ataques de dicionário às palavras-passe mais ingénias, que, mesmo convertidas em hash, são possíveis de encontrar bases de dados destas na internet.

relativos a *Empregados* e *Independentes* e o login de *Utilizadores ArtSOFT* só podem ser aceites se viverem encriptados.

Serviços relacionados:

- [ArtUsr/Login](#)
- [TerFch/Login](#)
- [GrhEmp/Login](#)
- [GrhInd/Login](#)

Apêndice L – Status WinSock

A seguir são listados os status winsock que poderão ser devolvidos pelos serviços, que poderão ser uma ajuda para determinar a causa de uma falha de operação.

Status	ID do status	Descrição
258	WAIT_TIMEOUT	Tempo máximo de espera atingido
10013	WSAEACCES	Não permitido acesso ao socket com as actuais permissões
10014	WSAEFAULT	Detectado endereço de pointer inválido numa chamada ao WinSock
10022	WSAEINVAL	An invalid argument was supplied
10024	WSAEMFILE	Número de sockets em uso excedeu o limite do sistema operativo
10038	WSAENOTSOCK	An operation was attempted on something that is not a socket
10049	WSAEADDRNOTAVAIL	O endereço pedido não é válido neste contexto
10050	WSAENETDOWN	O subsistema de rede não está operacional
10051	WSAENETUNREACH	A socket operation was attempted to an unreachable network
10052	WSAENETRESET	Ligaçāo perdida devido a uma falha detectada pelo 'keep-alive'
10053	WSAECONNABORTED	Ligaçāo cancelada na máquina local
10054	WSAECONNRESET	Ligaçāo cancelada de forma abortiva por iniciativa do contacto remoto
10101	WSAEDISCON	Ligaçāo cancelada de forma graciosa por iniciativa do contacto remoto
10055	WSAENOBUFS	Buffer de transmissão insuficiente
10056	WSAEISCONN	Este socket não suporta ListenPort
10057	WSAENOTCONN	Ligaçāo já foi encerrada
10058	WSAESHUTDOWN	Ligaçāo está a ser encerrada
10060	WSAETIMEDOUT	Falha na ligação ao contacto destino
10061	WSAECONNREFUSED	O contacto destino recusou esta operação
10064	WSAEHOSTDOWN	O contacto destino está inoperacional
10065	WSAEHOSTUNREACH	Não é possível localizar o contacto destino
10091	WSASYSNOTREADY	Falha no WSASStartup: sistema operativo,neste momento,não dispõe de recursos de rede
11001	WSAHOST_NOT_FOUND	Contacto destino desconhecido

Apêndice M – ArtCmdWS

O ArtCmdWs.dll permite aceder aos serviços XML do ArtSOFT através de aplicações desktop, efectuando as chamadas aos serviços directamente, sem usar um canal de comunicações, para além de poder utilizar serviços interactivos (exemplo: a impressão de documentos). Estão disponíveis todos os serviços disponibilizados pelo servidor XML e ainda os específicos deste utilitário.

Para assegurar um serviço muito utilizado pelo antigo ‘ArtCmd’, este DLL disponibiliza também o serviço ‘ArtCmd/ImportAscii’.

Função	Descrição
<code>long Startup(const char *nomeIni)</code>	Inicializa a plataforma ArtDB com uma empresa e ano de trabalho, através de um ficheiro de configuração passado como parâmetro. Se este não for indicado, será usado o nome ARTCMD.INI (Ver o formato deste ficheiro mais abaixo).
<code>long StartupEx(const char *PathFch, const char *NomeEmp, const char *Licenca, const char *PathForm, const char *Plugins, long dataTrabalho)</code>	Inicializa a plataforma ArtDB com uma empresa e ano de trabalho, através dos parâmetros passados: O caminho onde se encontra a base de dados da empresa, o nome do ficheiro da empresa, a licença ArtSOFT para uso da plataforma, o caminho onde se encontram os formulários de documentos, a lista de plugins ou DLL's extra a carregar, e a data de trabalho. Se esta não for indicada, será usada a data de sistema. Depois desta chamada, a plataforma mantém as tabelas de configuração abertas (ArtTBG e ArtTBE).
<code>void Shutdown()</code>	Esta função fecha as tabelas de configuração da empresa seleccionada.
<code>long Login(const char *user, const char *psw)</code>	Esta função permite efectuar o login de um utilizador à plataforma. Se o sistema que utiliza o ArtCmdWs pretender manter o sistema de acessos do ArtSOFT, deve pedir o nome e palavra-passe ao utilizador. Se isso não for desejado, deverá criar-se um utilizador específico para esse sistema (preferível), ou usar uma conta já existente.
<code>void *InitCmd()</code>	Esta função cria o contexto de execução (o objecto) que permite executar os serviços.
<code>cchar *Command(void *thisObj, const char *url, const char *XML)</code>	Esta função executa serviços. O parâmetro <code>thisObj</code> é obtido através da função <code>InitCmd</code> . O parâmetro <code>'url'</code> deve indicar qual o recurso ou serviço pretendido. O parâmetro XML deve conter uma string XML com os parâmetros necessários. Esta função retorna um apontador para uma área de memória que contém a string XML com a resposta.
<code>void Release(void *Obj)</code>	Esta função permite destruir o contexto de execução de serviços, libertando os recursos utilizados.

Quando a aplicação utilizadora pretender utilizar a base de dados de uma empresa, deve chamar as funções Startup ou StartupEx que irá abrir as tabelas de configuração da empresa. Após terminar o uso dessa empresa, deve chamar-se Shutdown, encerrando as tabelas de configuração.

Após efectuar Startup(Ex) com sucesso, deve efectuar-se o login de um utilizador real ou específico da interligação. Se o login não for executado com sucesso, não será possível chamar a função InitCmd.

Após ter efectuado o login com sucesso, pode efectuar-se um ciclo de chamadas às funções InitCmd / Command / Command / ... / Command / Release.

Sempre que forem executados uma série de comandos, deve fazer-se InitCmd, seguido do conjunto de comandos pretendido, e por fim, deve chamar-se Release. Deve ter-se em conta o seguinte:

- Não é possível chamar ‘Command’ sem InitCmd, pois este necessita de um objecto válido para executar o comando.
- Deve evitar-se a sequência ‘InitCmd / Command / Release’, pois o comando ‘Release’ limpa a cache de objectos usados, que deixa de poder ser reutilizada em futuros comandos, perdendo-se eficiência.
- Deve evitar-se fazer InitCmd uma única vez no arranque, e Release apenas no fim da aplicação, porque isso irá manter todos os objectos utilizados em cache, levando a um consumo exagerado de memória, e consequentemente, pode perder-se eficiencia em sistemas com menos memória disponível.

O formato do ficheiro de configuração deve conter:

Conteúdo	Descrição
[CONFIG]	
PathFch =C:\Art\FICH	O caminho onde se encontra a base de dados da empresa
NomeEmp =Demonstracao	O nome do ficheiro da empresa
PathForm=C:\Art\Forms	O caminho onde se encontram os formulários de documentos
Plugins =DocPrintCmd	A lista de plugins ou DLL's extra a carregar
Licenca =BEBEBEBECAFE1234ABCDEF ... 12FA	A licença ArtSOFT para uso da plataforma ArtCMD.
Year =2090	Ano de trabalho. Serve para controlar qual o ano da empresa e data de trabalho a usar. Se a data de sistema for maior que a data de encerramento fiscal da empresa para esse ano, a data de trabalho

passará a ser essa. Ex: Year=2090, Data de sistema=2091-01-15, data de trabalho: 2090-12-31.

O ficheiro de configuração deverá conter o ano de trabalho ('Year'), Assumindo a data de trabalho como sendo a data de sistema, excepto nas condições descritas acima. Se for necessário ter controlo sobre a data de trabalho, recomenda-se a inicialização através da função StartupEx.

Interface ArtCmdWsDotNet

Para os utilizadores que pretendam usar o ArtCmdWS através da plataforma .NET, foi disponibilizada uma assembly de nome '**ArtCmdWsDotNet.dll**', que trata dos pormenores de mais baixo nível.

Este DLL deve ser colocado no mesmo local onde estiverem os restantes DLL e / ou executável da aplicação.

Apresenta-se aqui a classe gerada a partir do 'metadata' do DLL:

```
using System;
namespace ArtCmdWsDotNet
{
  public class ArtCmdWS : IDisposable
  {
    public ArtCmdWS();
    public StartupErrors Startup(string PathIniCfg);
    public StartupErrors StartupEx(string PathArtCmd, string PathFch, string NomeEmp,
      string Licenca, string PathForm, string Plugins, int dataTrabalho);
    public void Shutdown();
    public LoginErrors Login(string user, string psw);
    public string Command(string url, string XML);
    public void Release();
  }
}
```

Como pode verificar, este interface é similar ao descrito atrás, excepto:

- Em StartupEx, existe mais um parâmetro que deve informar qual o caminho onde se encontram os DLL do ArtDB/ArtSOFT. Se esses DLL estiverem no mesmo local onde for executada a aplicação utilizadora (executável de arranque), indicar "", senão, deve indicar-se o local exacto onde estes de encontram, permitindo chamar 'SetDllDirectory'. Se isso não for feito, este interface não vai conseguir carregar os DLL necessários e devolve 'StartupErrors.StartArtCmdLoadError'.
- Se a biblioteca for iniciada via Startup, a 'PathIniCfg' deve estar no mesmo directório que os DLL do ArtDB/ArtSOFT, sendo obtido o directorio dos DLL a partir do caminho e nome deste ficheiro.
- Command não necessita da chamada prévia a InitCmd. Isto é feito automaticamente pelo objecto antes da primeira chamada, guardando internamente o handle 'thisObj' necessário, até que seja destruído por Release.
- Logo que não seja necessário executar mais comandos de imediato, deve chamar-se Release, para que o ArtDB limpe a cache dos objectos que utilizou e liberte a memória usada. A falha da chamada a esta função ou da função ShutDown não trás qualquer problema, pois o destructor deste objecto encarrega-se dessa tarefa, só que apenas no fim da aplicação. Isto não é aconselhável, pois poderá reduzir a eficiência da aplicação utilizadora, provocando consumos de memória exageradamente altos.

Apêndice N – Controlo de versões

Versão	Data	Descrição
V1.0	2010-09-10	Adicionado o controlo de versões
V1.1	2010-09-21	Alterada a descrição do serviço ArtDB/_WSList e melhorias na descrição do ArtCmdWS
V1.2	2011-01-19	Novo Serviço ArtDB/Cambios, introdução de credenciais em Queries/SQLQuery, melhorias na descrição do ArtUsr/Login, e updates no ArtCmdWS.
V1.3	2011-10-01	Novos serviços , DocFch/DocUpdate, DocFch/DocSignature, CRMHdr/GetQualif, GrhEmp/UpdEventCad, GrhEmp/DelEventCad, CtaFch/UpdateAccounts, CtaFch/DeleteAccounts, CtaLan/LanUpdate, CtaLan/LanDelete. Alterações nas descrições do CRMHdr/Update, CRMHdr/Delete.
V1.4	2013-09-16	Adicionadas novas funções de XMLReports: LogicalAnd e LogicalOR