



Engage[®] Business Solution 5

Configuración de Documentos
Electrónicos en Engage 5.8 y
Superiores



Contenido

ACERCA DE ENGAGE® BUSINESS SOLUTION 5	3
ALCANCE DE ESTE DOCUMENTO	3
CADUCIDAD DE ESTE DOCUMENTO	3
CARACTERÍSTICAS GENERALES	4
DEFINICIÓN DEL TEMPLATE EN DESIGNER	4
INVOCACIÓN AL SERVICIO DE GENERACIÓN DE DOCUMENTOS.....	6
MENSAJE DE SALIDA	7
ENVÍO DE EMAILS CON TEMPLATES DOCX	9
ENVÍO DE EMAILS CON ADJUNTOS DINÁMICOS	10
CONFIGURACIÓN DE ENGAGE INTEGRATION SERVICE	12
REEMPLAZOS DE CONTENIDO	13
FECHA - HORA	13
NÚMEROS.....	14
COMENTARIOS, ESTILOS Y PROPIEDADES DE WORD	14
IMÁGENES	15
COMANDOS.....	16
CURSORES.....	19
GENERAR DOCUMENTO COMO ADJUNTO	22

Acerca de Engage® Business Solution 5

Engage® Business Solution 5 es una plataforma tecnológica orientada al diseño e implementación de procesos y aplicaciones de negocio de variada naturaleza.

Su organización modular permite instalar, configurar y distribuir sus componentes de muchas maneras posibles, posibilitando un escalamiento horizontal y vertical de la instalación.

Alcance de este documento

Describir las funcionalidades provistas por el componente de documentos electrónicos (Engage Doc Services) y configuraciones necesarias para la generación de documentos PDF, DOCX y HTML a partir de templates DOCX o TXT, así como también el uso de este tipo de templates para el envío de emails y adjuntos con reemplazos dinámicos.

Si tiene dudas o inquietudes acerca del presente documento, por favor, sírvase remitirlas a:
supportcrm@soluciones-ar.com.ar

Caducidad de este documento

Fecha de última actualización del documento: 07 de Julio de 2020.

Fecha de caducidad del documento: 31 de Diciembre de 2020.

Una vez que haya caducado el presente documento, remitirse al sitio o al correo de soporte oficiales para obtener una nueva versión del mismo.

Características Generales

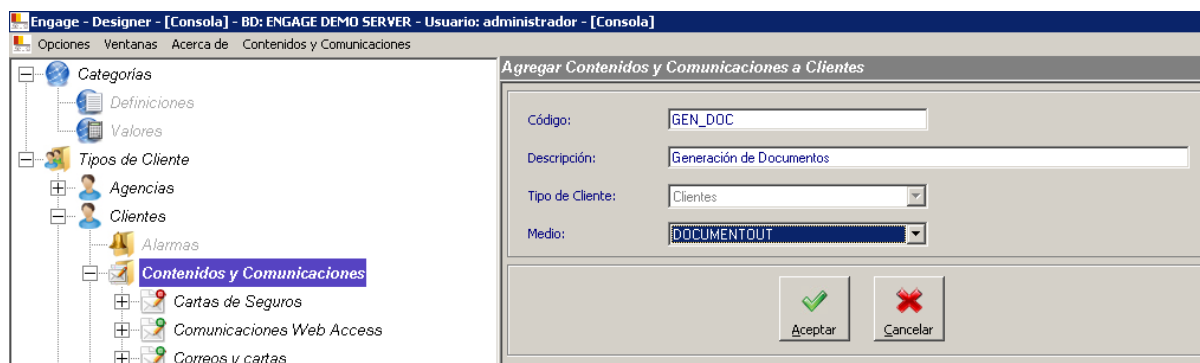
A partir de la **versión 5.8.0.0** de Engage se introduce un nuevo componente de documentos electrónicos y un sistema de templates para la generación dinámica de documentos en formatos **PDF, DOCX y HTML**. Los templates se pueden armar en formato DOCX o TXT y existen diversos tipos de reemplazos que se pueden utilizar y que serán explicados a lo largo del documento. Esta nueva funcionalidad también permite el envío de emails utilizando este tipo de templates.

Principales Ventajas del nuevo componente:

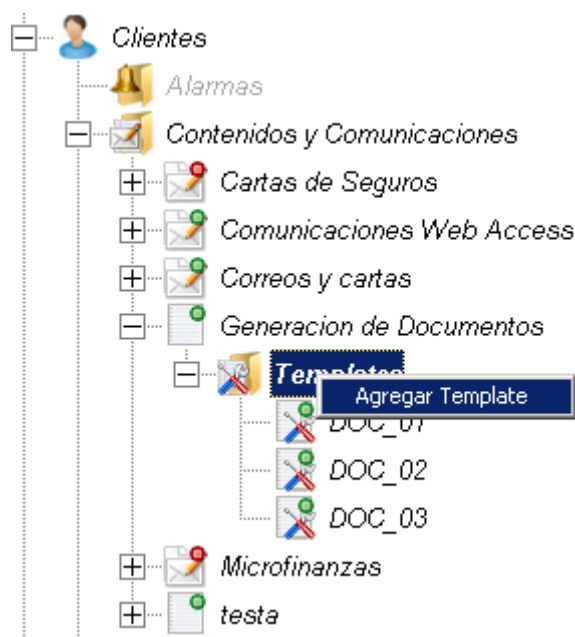
- Generación de documentos **PDF** de forma nativa.
- Generación de Reportes con saltos de página, encabezado y pie de página fijos.
- Utilización de cursores para realizar consultas dentro del documento fuente.
- Ahora se utiliza directamente un documento MS Word como template, sin necesidad de convertirlo en HTML. Modificar un template existente ya no implica tener que modificar código HTML o modificar el documento WORD original teniendo que volver a generar el HTML y editarlo, simplemente se guarda el documento con extensión DOCX y se utiliza como template fuente.
- Pueden utilizarse todos los objetos, estilos y propiedades que provee Word. Estos serán tenidos en cuenta a la hora de generarse el documento (tablas, colores, fuentes, alineaciones, etc).
- Utilización de reemplazos dentro de comentarios.
- Envío de emails y adjuntos con reemplazos dinámicos a partir de templates DOCX.

Definición del Template en Designer

Para definir los nuevos tipos de documentos electrónicos en Designer, agregar un nuevo grupo de Contenidos y Comunicaciones con **Medio = "DOCUMENTOUT"**, tal como se ve en la siguiente imagen:



Luego, presionar botón derecho sobre el grupo creado y seleccionar la opción "Agregar Template":



Agregar Templates a Generacion de Documentos

Tipo de proceso:	Generacion de Documentos	
Código de Actividad:	DOC_04	
Descripción:	Documento 4	
Template Docx:	es (x86)\Engage\W5\EngageApplicationServices\Documents\Input\sample1.docx	...
Template Text:	gram Files (x86)\Engage\W5\EngageApplicationServices\Documents\Input\text.txt	...

 **Aceptar**
 **Cancelar**

Código de Actividad: Código del template que luego se utilizara para la generación del documento.

Descripción: Descripción del template.

Template Docx: Indicar la ruta donde se encuentra el template DOCX que se desea utilizar.

Template Text: Indicar la ruta donde se encuentra el template TXT (si es que se desea utilizar).

Invocación al servicio de Generación de Documentos

Para poder invocar al servicio y de esta manera poder generar un documento a partir de un template .DOCX o .TXT se debe crear una transacción de socket en Designer con el siguiente mensaje de entrada:

DOCX|CustPkey|JobPkey|UserId|CallTypeCode|DocxOutputFormat[EOM]

Dónde:

DOCX: (Obligatorio y Fijo) Es el tipo de servicio.

CustPkey: Pkey del cliente.

JobPkey: Pkey del proceso.

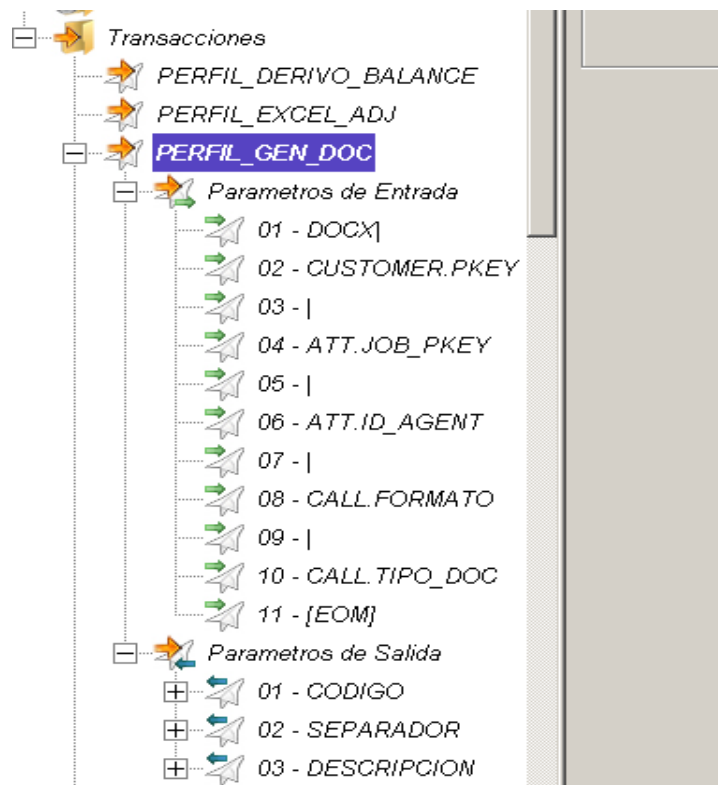
UserId: Usuario de Engage.

CallTypeCode: Código de actividad del template (En el ejemplo de la imagen anterior seria "DOC_04").

DocxOutputFormat: Tipo/Formato de documento que se quiere generar. Los valores posibles son PDF, DOCX y HTML, dependiendo del tipo de documento que se desea generar.

[EOM]: (Obligatorio) Delimitador de fin de mensaje.

Ejemplo:



Dónde:

- CALL.FORMATO es un campo de la entidad del trámite donde se guarda el código de actividad del template en base al cual se desea generar el documento.
- CALL.TIPO_DOC es un campo de la entidad del trámite donde se guarda el tipo de documento que se desea generar. PDF, DOCX o HTML.

Como se puede observar en la imagen anterior, los parámetros de salida de la transacción de socket son los mismos que cuando se consume un Web Service o se envía un email: Código, Separador y Descripción.

Mensaje de Salida

El formato del mensaje de salida del servicio de documentos de Engage Integration Service es el siguiente:

VALUE|Path_Archivo_Salida_Template_Docx|Path_Archivo_Salida_Template_Txt

Esto es así porque un template Docx no puede tener una salida a texto (sólo puede ser Docx, Html o Pdf), y un template Txt sólo puede tener una salida a texto.

Con lo cual, si en Designer se utilizan las dos alternativas de templates, es decir, si se informa un template Docx y uno Txt, entonces necesariamente habrá dos salidas. No es como en los mails, en los que se utiliza Html y Txt en forma no excluyente, y un .eml puede contener ambas alternativas de cuerpo.

A continuación se provee un ejemplo de una función SQL para parsear un string, por si se deseara obtener la ruta donde se generó el documento. Los parámetros de la función son, el string a parsear, el carácter delimitador, y el índice del campo que se desea que te devuelva.

```
CREATE FUNCTION [dbo].[fnSplitString]
(
    @string VARCHAR(MAX),
    @delimiter CHAR(1),
    @index INT
)
RETURNS VARCHAR(MAX)
BEGIN
    DECLARE @start INT, @end INT, @pos INT, @result VARCHAR(MAX)
    SELECT @start = 1, @end = CHARINDEX(@delimiter, @string), @pos = 1,
    @result = ''
    WHILE @start < LEN(@string) + 1 BEGIN
        IF @end = 0
            BEGIN
                SET @end = LEN(@string) + 1
            END
        IF @pos = @index
            BEGIN
                SET @result = SUBSTRING(@string, @start, @end -
                @start)
                SET @start = LEN(@string) + 1
            END
        ELSE
            BEGIN
                SET @start = @end + 1
                SET @end = CHARINDEX(@delimiter, @string, @start)
                SET @pos = @pos + 1
            END
        END
    RETURN @result
END
```


Envío de Emails con Templates DOCX

Para el envío de emails utilizando templates armados con Word (.docx), se deben seguir exactamente los mismos pasos que los que se venían realizando para el envío de emails con templates HTML, con la única diferencia que en el campo "Modelo Origen" del formato de email, se debe indicar la ruta del template .DOCX que se desea utilizar (en lugar de un archivo .html), tal como se ve en las siguientes imágenes:

Formatos

- Apertura de Sumario (cambiar)
- Apertura de Sumario (Financiero)
- Carta de acuerdo
- Carta de Deuda Modelo 1
- CARTA SEGUROS
- Cotización de servicio
- Detalle de deuda
- Email Amazon
- Envío de Documentos

Modificar Envío de Documentos de Correos y cartas

Tipo de proceso: Correos y cartas
 Código de Actividad: EMAIL_DOCUMENTOS
 Descripción: Envío de Documentos

Modelo Origen: es (x86)\Engage\W5\EngageApplicationServices\Email\Pickup\Documentos.docx
 Modelo Alternativo:
 Archivos adjuntos: TMT_SOPORTE.RUTA
 Nombre Remitente: Soporte Engage Dirección Remitente: soporte.engage@gmail.com
 CC: prueba@gmail.com BCC: prueba2@gmail.com
 Asunto: Envío de documento

Destino
☒ Predefinido Tipo de Cliente CS_DATOS_CLIENTE.EMAIL
☐ Personalizado
 Responder a:
 Aceptar Cancelar

Modificar Envío de Documentos de Correos y cartas

Tipo de proceso: Correos y cartas
 Código de Actividad: EMAIL_DOCUMENTOS
 Descripción: Envío de Documentos

Modelo Origen: es (x86)\Engage\W5\EngageApplicationServices\Email\Pickup\Documentos.docx
 Modelo Alternativo:
 Archivos adjuntos: TMT_SOPORTE.RUTA
 Nombre Remitente: Soporte Engage Dirección Remitente: soporte.engage@gmail.com
 CC: prueba@gmail.com BCC: prueba2@gmail.com
 Asunto: Envío de documento

Destino
☒ Predefinido Tipo de Cliente CS_DATOS_CLIENTE.EMAIL
☐ Personalizado
 Responder a:
 Aceptar Cancelar

Envío de Emails con adjuntos dinámicos

Los templates DOCX con reemplazos dinámicos pueden ser utilizados, además de en el cuerpo del email, en los archivos que se envían como adjunto, para esto, al definir el formato de email, debe indicarse un documento con extensión .DOCX en el campo "Archivos adjuntos", tal como se ve en la siguiente imagen:

Modificar Mail de Documentos de Correos y cartas

Tipo de proceso: Correos y cartas

Código de Actividad: MAIL_GEN_DOC

Descripción: Mail de Documentos

Modelo Origen: les (x86)\Engage\W5\EngageApplicationServices\Documents\Input\5-SEAT.docx

Modelo Alternativo:

Archivos adjuntos: les (x86)\Engage\W5\EngageApplicationServices\Documents\Input\5-SEAT.docx

Nombre Remitente: Soporte Engage Dirección Remitente: soporte.engage@gmail.com

CC: prueba@gmail.com BCC:

Asunto: Prueba Documentos

Destino

☒ Predefinido Tipo de Cliente CS_DATOS_CLIENTE.EMAIL

☐ Personalizado

Responder a:

Aceptar Cancelar

Además, para que se acepten reemplazos dinámicos en el template deben agregarse los prefijos "(doc)" , "(docx)" o "(pdf)" delante del path al archivo para marcarlo como dinámico, lo cual también determinará la extensión que tendrá el nombre del archivo una vez que se adjunte al email:

Modificar Mail de Documentos de Correos y cartas

Tipo de proceso: Correos y cartas

Código de Actividad: MAIL_GEN_DOC

Descripción: Mail de Documentos

Modelo Origen: les (x86)\Engage\W5\EngageApplicationServices\Documents\Input\5-SEAT.docx

Modelo Alternativo:

Archivos adjuntos: (docx) C:\Program Files (x86)\Engage\W5\EngageApplicationServices\Documents

Nombre Remitente: Soporte Engage Dirección Remitente: soporte.engage@gmail.com

CC: prueba@gmail.com BCC:

Asunto: Prueba Documentos

Destino

☒ Predefinido Tipo de Cliente CS_DATOS_CLIENTE.EMAIL

☐ Personalizado

Responder a:

Aceptar Cancelar

En el caso de la imagen anterior se enviará un email con un archivo adjunto con reemplazos dinámicos y extensión .docx.

Modificar Mail de Documentos de Correos y cartas

Tipo de proceso:	Correos y cartas		
Código de Actividad:	MAIL_GEN_DOC		
Descripción:	Mail de Documentos		
Modelo Origen:	C:\Program Files (x86)\Engage\W5\EngageApplicationServices\Documents\Input\	...	
Modelo Alternativo:		...	
Archivos adjuntos:	(pdf) C:\Program Files (x86)\Engage\W5\EngageApplicationServices\Documents\I	...	
Nombre Remitente:	Soporte Engage	Dirección Remitente:	soporte.engage@gmail.com
CC:	prueba@gmail.com	BCC:	
Asunto:	Prueba Documentos		
Destino			
☒ Predefinido Tipo de Cliente		CS_DATOS_CLIENTE.EMAIL	
☐ Personalizado			
Responder a:			




En el caso de la imagen anterior se enviará un email con un archivo adjunto con reemplazos dinámicos y extensión .pdf.

Configuración de Engage Integration Service

En el archivo de configuración de EIS (EngageIntegrationService.xml) se deberán configurar los siguientes parámetros:

```
<Documents>
  <OutputFolder UseNamedSubFolders="False" NumberedSubFoldersCount="0">C:\Program Files
(x86)\Engage\V5\EngageApplicationServices\Documents\Output</OutputFolder>
  <PropertiesFilePath>.\doc4j.properties</PropertiesFilePath>
  <UseIkv>True</UseIkv>
  <!-- JavaExePath>C:\Program Files (x86)\Java\jre7</JavaExePath -->
  <!-- JavaBuildHtmlTimeout>30</JavaBuildHtmlTimeout -->
  <!-- JavaBuildPdfTimeout>30</JavaBuildPdfTimeout -->
</Documents>
```

OutputFolder: Ruta de la carpeta donde se desea que se guarden los archivos generados.

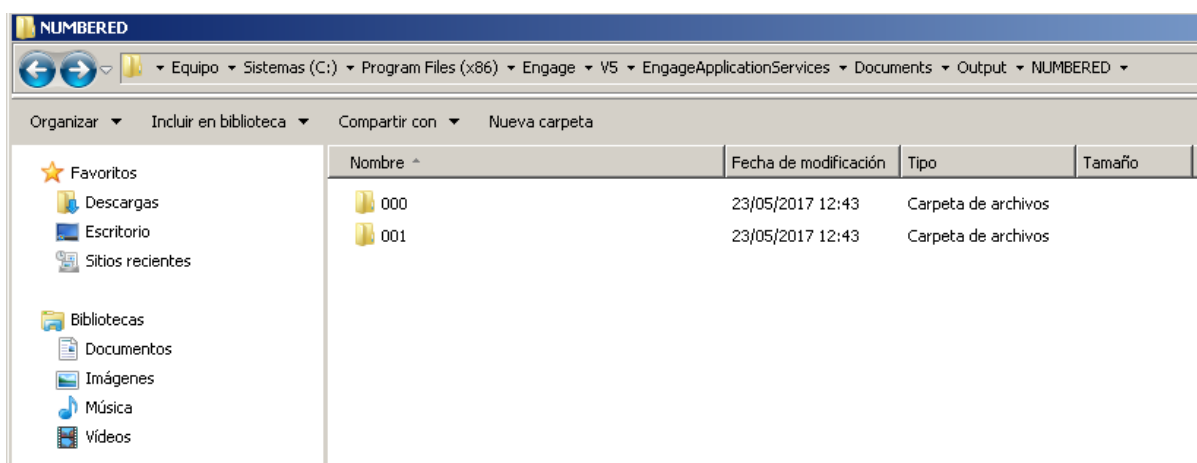
UseNamedSubFolders: Valores posibles: True y False.

Utilizar el valor True si se desea que los documentos generados se guarden en una subcarpeta con un nombre especificado dentro del template. Para especificar el nombre de la carpeta se pueden utilizar "Comandos" (Ver sección "Comandos" de este documento).

Utilizar el valor false si se desea que los documentos se guarden en la ruta definida en la sección "OutputFolder" sin ninguna subcarpeta.

Valor por defecto: False.

NumberedSubFoldersCount: Al utilizar este parámetro, los documentos se generaran en subcarpetas numeradas. La cantidad de carpetas numeradas a utilizar depende del número definido en este parámetro. El valor 0 implica que no se utilizaran carpetas numeradas. Ejemplo, si el parámetro se setea con el valor 2, las subcarpetas se verán de la siguiente forma:



Como se observa en la imagen anterior, se utilizan dos subcarpetas numeradas (porque el parámetro "NumberedSubFoldersCount" se definió con el valor 2) y los archivos se van a ir generando de forma alternada entre una carpeta y la otra.

Valor por defecto: 0.

Nota: Para ver las distintas combinaciones de valores posibles para estos parámetros, ver la sección "Comandos" de este documento.

Reemplazos de Contenido

Los reemplazos variables dentro de los templates se deben indicar entre llaves dobles y con el formato TABLA.CAMPO:

```
{{ TABLA.CAMPO }}
```

A diferencia del modelo anterior (templates HTML) donde los reemplazos de contenido se informaban de la forma

```
<ENG_TABLA.CAMPO>
```

Ejemplo de reemplazo:

```
{{ C_CUSTOMER_DATA.NOMBRE }}
```

Donde C_CUSTOMER_DATA es el nombre de la tabla y NOMBRE es el nombre del campo que se desea que se reemplace al momento de generarse el documento o enviarse un email utilizando un template DOCX.

Fecha - Hora:

Para utilizar un reemplazo con la fecha y hora actual se puede utilizar la variable DATETIME entre llaves dobles, ejemplo:

Fecha Actual: {{ DATETIME }}

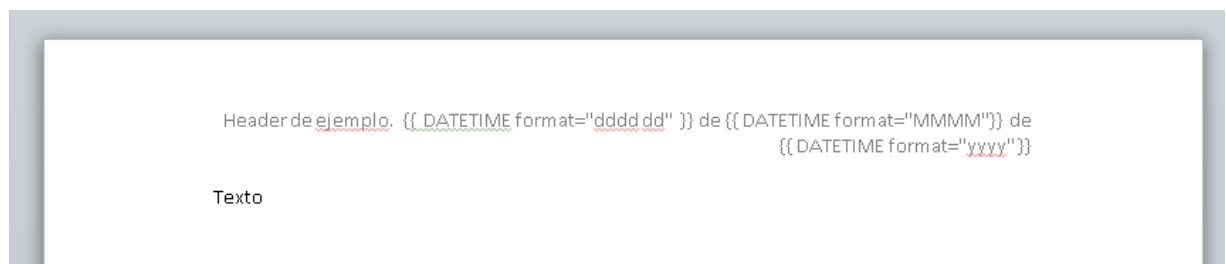
También se le puede dar formato a la fecha utilizando el parámetro "FORMAT", ejemplo:

Datetime con formato: {{ DATETIME format="dd/MM/yyyy" }}

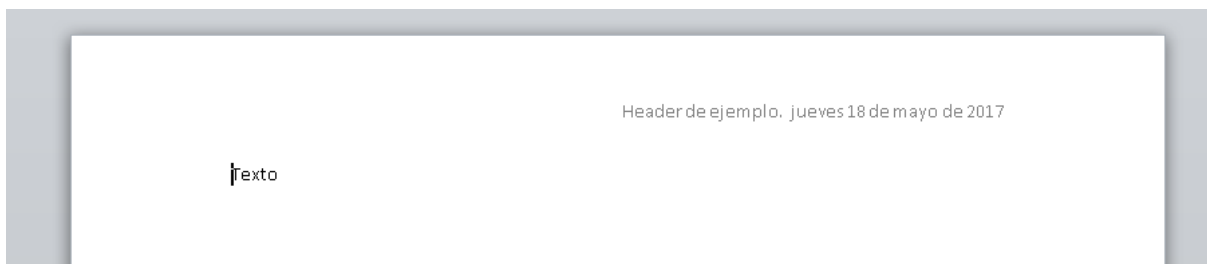
Esto nos generará una fecha con el formato dd/mm/yyyy

Ejemplo de uso de Fechas en el header de un documento con otro formato:

Template:



Resultado:



Números:

Para los números se pueden utilizar reemplazos estándar ({{TABLA.CAMPO}}), ejemplo:

`{{ C_CUSTOMER_DATA.MORA_TOTAL }}`

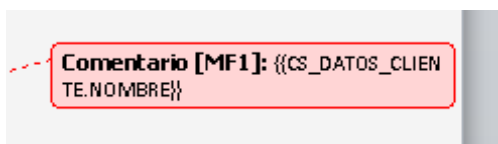
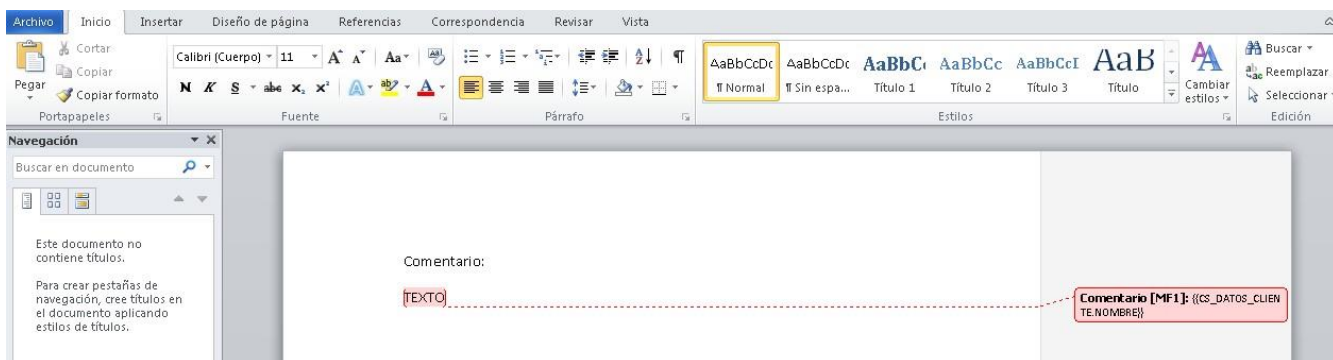
O bien se les puede dar formato utilizando el parámetro format, ejemplo:

`{{ C_CUSTOMER_DATA.MORA_TOTAL format="0.0000" }}`

Esto hará que los números se muestren con 4 decimales.

Comentarios, estilos y propiedades de Word:

Utilizando templates DOCX se puede hacer uso de reemplazos dentro de los comentarios que provee Microsoft Word, por ejemplo, en el siguiente template, la palabra TEXTO se reemplazara por el contenido del campo NOMBRE de la tabla CS_DATOS_CLIENTE cuando se genere el documento.



También se puede hacer uso de los estilos que provee el Word, ejemplo, si utilizamos el siguiente tipo de reemplazo:

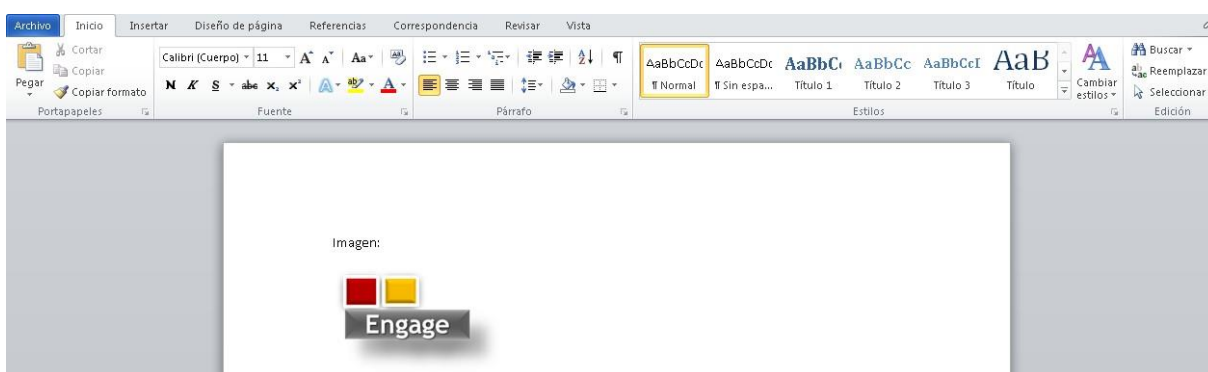
`{{ C_CUSTOMER_DATA.NOMBRE }}`

Cuando se reemplace el campo nombre por su valor, aparecerá en color rojo en el documento generado.

Las propiedades que provee el Word como por ejemplo Texto alineado, justificado, centrado, etc también pueden ser utilizadas en el template y serán tomadas en cuenta a la hora de generarse el documento.

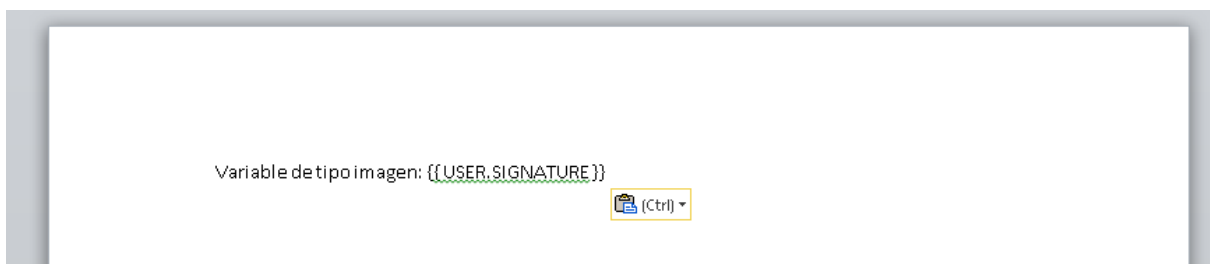
Imágenes:

Para mostrar imágenes, se pueden incluir normalmente en el Word, tal como se ve en el siguiente ejemplo:

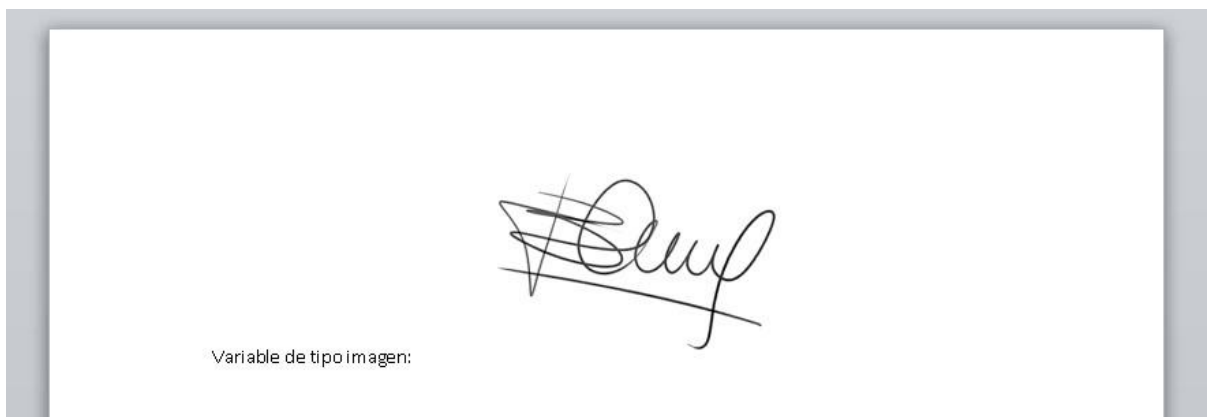


O bien se puede mostrar la firma del usuario de Engage (imagen que se carga desde el módulo Security), haciendo uso del reemplazo "USER.SIGNATURE", ejemplo:

Template:



Documento Generado:



Además, se pueden utilizar los parámetros WIDTH y HEIGHT para darle el ancho y el alto deseado al reemplazo de la imagen, ejemplo:

```
{{ USER.SIGNATURE WIDTH="200" height="200" }}
```

Esto hará que la imagen se muestre con un ancho y un alto de 200 píxeles.

Comandos:

Se pueden utilizar comandos dentro del template para indicar el nombre de la subcarpeta dentro de la cual se va a generar el documento (en caso de que no se utilice este comando, los documentos se generaran en la ruta especificada en el archivo de configuración del EIS sin ninguna subcarpeta asociada). Para que este comando funcione, el parámetro "UseNamedSubFolders" del archivo de configuración del EIS debe estar en "True".

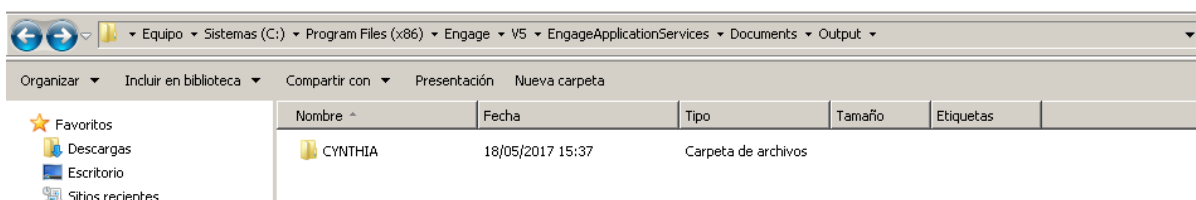
```
{# SET_PARAMETER NAME="OUTPUT_SUBFOLDER" SOURCEDATA="USER.NAME" #}
```

El parámetro sourcedata indica desde que Tabla/Campo se obtendrá el valor del nombre con el cual se creará la subcarpeta en la que se generarán los documentos.

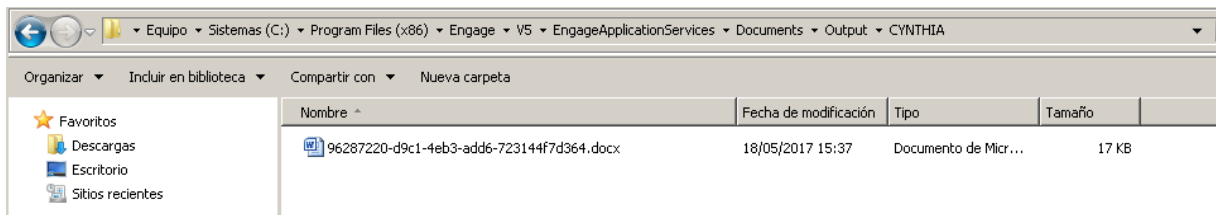
Ejemplo:

```
{# SET_PARAMETER NAME="OUTPUT_SUBFOLDER" SOURCEDATA="USER.NAME" #}
```

Nos creara una subcarpeta con el nombre del usuario (en este caso de ejemplo el usuario se llama CYNTHIA).



Y dentro de la misma estará el documento DOCX, PDF o HTML generado:

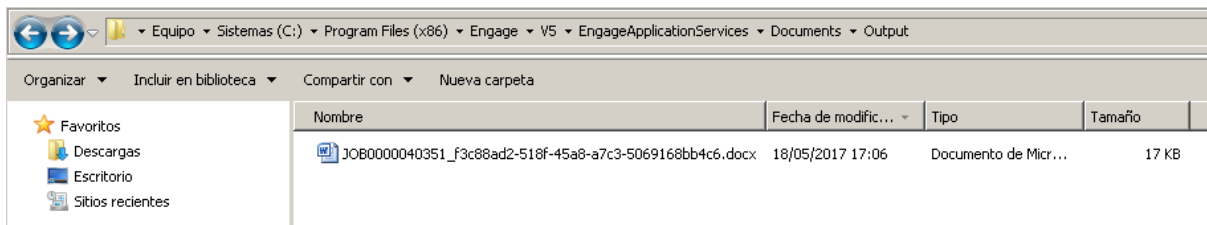


También se puede utilizar un comando para que el nombre de los documentos se genere con un prefijo, por ejemplo:

```
{# SET_PARAMETER NAME="OUTPUT_FILE_NAME" SOURCEDATA="PHYSICAL_JOB.JOB_SEQ"
FORMAT="JOB'0000000000" #}
```

El parámetro SOURCEDATA indica el campo del cual se va a obtener el valor para incluir como prefijo en el nombre del archivo. En este caso de ejemplo el campo JOB_SEQ.

El parámetro FORMAT es opcional y sirve para darle el formato que necesitamos. En este caso de ejemplo el prefijo se compondrá del string "JOB" seguido de 10 números que se reemplazaran con el job_seq obtenido del sourcedata, completando el resto de los números con 0. Ejemplo de documento generado con el comando del ejemplo:



En la imagen anterior se puede observar que el prefijo del nombre del documento generado está compuesto por el string "JOB" seguido del job_seq "40351" completando el resto de los números con el valor 0.

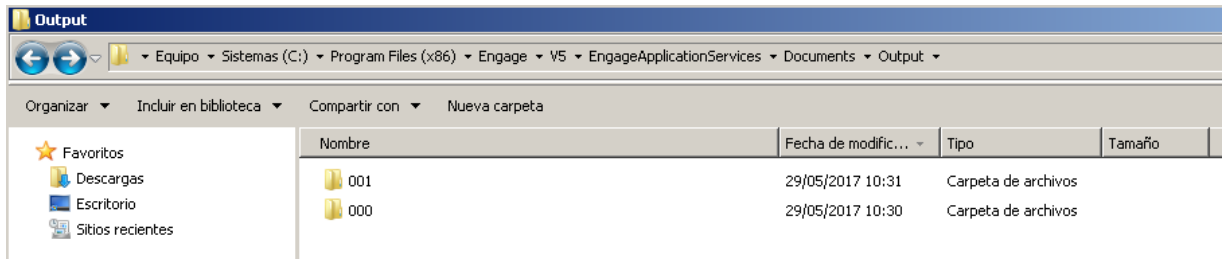
Comportamientos de acuerdo a las distintas combinaciones entre el parámetro "OUTPUT_SUBFOLDER" y los parámetros del archivo de configuración de EIS "NumberedSubFoldersCount" y "UseNamedSubFolders":

- Si UseNamedSubFolders es False y NumberedSubFoldersCount es 0 entonces los archivos generados se guardaran en la carpeta raíz definida en el nodo "OutputFolder" del archivo de configuración de EIS:
- Si UseNamedSubFolders es False y NumberedSubFoldersCount es mayor a 0 entonces los archivos generados se guardaran de la siguiente forma:

Carpeta Raíz + Carpeta Numerada

Creándose tantas carpetas numeradas como se hayan definido en el parámetro NumberedSubFoldersCount.

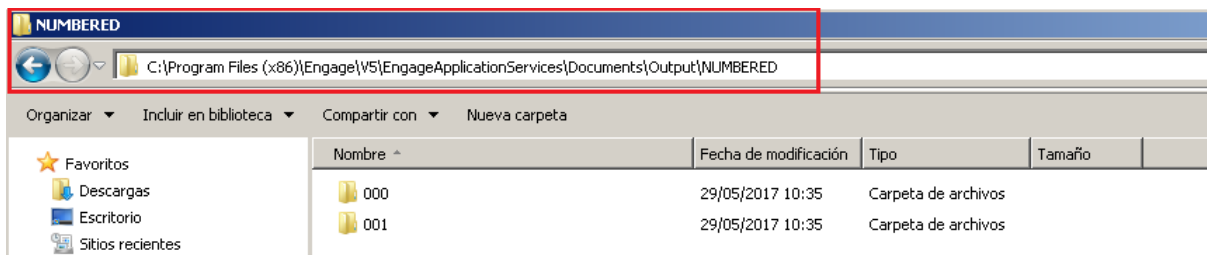
Ejemplo (con el parámetro NumberedSubFoldersCount en 2):



- Si UseNamedSubFolders es True, no se utilizó el parámetro "OUTPUT_SUBFOLDER" en el template y NumberedSubFoldersCount es mayor a 0 entonces los archivos generados se guardaran de la siguiente forma:

Carpeta Raíz + "\NUMBERED\" + Carpeta Numerada

Ejemplo:



- Si UseNamedSubFolders es True, no se utilizó el parámetro "OUTPUT_SUBFOLDER" en el template y NumberedSubFoldersCount es igual a 0 entonces los archivos generados se guardaran en la carpeta raíz definida en el nodo "OutputFolder" del archivo de configuración de EIS:

- Si UseNamedSubFolders es True, se incluyó el parámetro "OUTPUT_SUBFOLDER" en el template y NumberedSubFoldersCount es mayor a 0 entonces los archivos generados se guardaran de la siguiente forma:

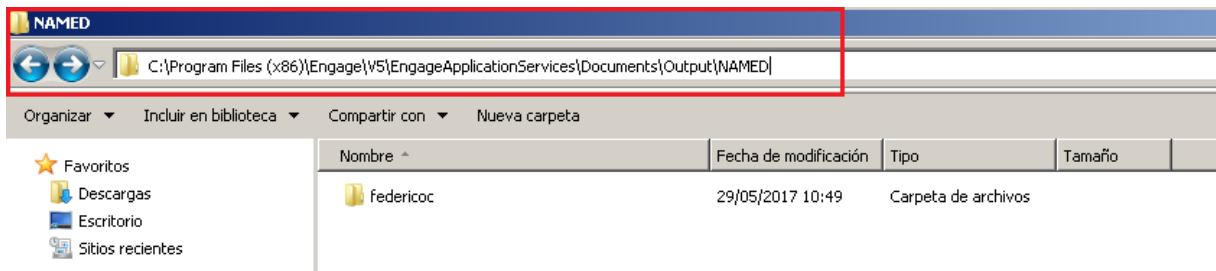
Carpeta Raíz + "\NAMED\" + OUTPUT_SUBFOLDER

Ejemplo:

Template:

```
{# SET_PARAMETER NAME="OUTPUT_SUBFOLDER" SOURCEDATA="USER.NAME" #}
```

Carpeta generada con el nombre de usuario:



- Si UseNamedSubFolders es True, se incluyó el parámetro "OUTPUT_SUBFOLDER" en el template y NumberedSubFoldersCount es igual a 0 entonces los archivos generados se guardaran de la siguiente forma:

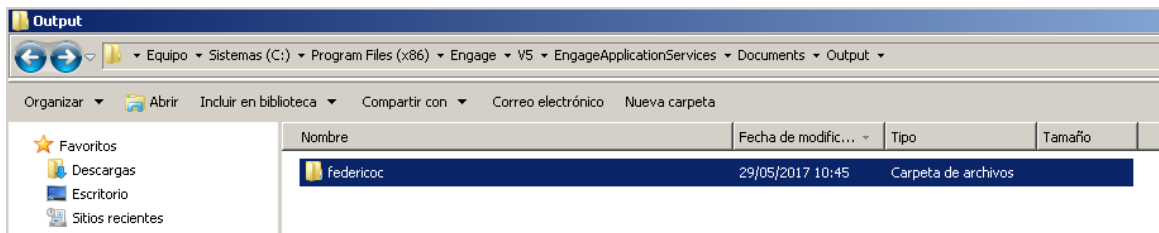
Carpeta Raíz + OUTPUT_SUBFOLDER

Ejemplo:

Template:

```
{# SET_PARAMETER NAME="OUTPUT_SUBFOLDER" SOURCEDATA="USER.NAME" #}
```

Carpeta generada con el nombre de usuario:



Cursores:

El nuevo sistema de templates DOCX soporta de forma nativa el uso de cursores.

-Para definir un cursor se utiliza el comando OPEN_CURSOR. El parámetro "NAME" es obligatorio.

Para ejecutar en un documento electrónico un Cursor con variables dinámicas hay que utilizar las siguientes palabras clave en la sentencia SQL:

FilterColumn: Nombre de la columna de la tabla o del sub-query mencionado en el From que se usará como filtro.
FilterParameter: CustPkey, JobPkey, UserID, CampaignPkey (este último sólo disponible en emails de campañas)

Las variables que se pueden usar son las que se pasan por parámetro en el mensaje a EIS.

Ejemplo de uso:

Mensaje a EIS:

```
DOCX|0003916516_77|1fdd5467-216c-41f4-959b-43222998c8da|userid|ACT  
DOCX|_TEST_DOCUMENT|DOCX[EOM]
```

Definición del cursor en el template:

```
{# OPEN_CURSOR NAME="clientes"
  COLUMNS="NOMBRE,PRIMER_APELLIDO,FECHA_ALTA"
  FROM="C_CUSTOMER_DATA as c"
  WHERE="c.TIPO_DOC=50 AND OFICIAL='0996'"
  ORDERBY="c.nombre DESC"
  FILTERCOLUMN="PAR_KEY"
  FILTERPARAMETER="CUSTPKEY" #}
```

Query ejecutado:

```
SELECT NOMBRE,PRIMER_APELLIDO,FECHA_ALTA FROM C_CUSTOMER_DATA as c WHERE
PAR_KEY='0003916516_77' AND c.TIPO_DOC=50 AND OFICIAL='0996'
ORDER BY c.nombre DESC
```

Nótese que en el query se agrega automáticamente el fragmento PAR_KEY='0003916516_77' en el WHERE.

Si se necesita por ejemplo, obtener descripciones de categorías o hacer joins, se puede armar un subquery en el FROM, por ejemplo como los siguientes:

Ejemplo 1:

```
{# OPEN_CURSOR NAME="clientes" COLUMNS="NOMBRE" FROM="(select c.nombre AS NOMBRE,c.PAR_KEY AS
PAR_KEY from CS_DATOS_CLIENTE c INNER JOIN PHYSICAL_JOB P ON C.PAR_KEY=P.PKEY_CUSTOMER) T
"WHERE="1=1" FILTERCOLUMN="PAR_KEY" FILTERPARAMETER="CUSTPKEY" #}
```

Ejemplo 2:

```
{# OPEN_CURSOR NAME="reasons" COLUMNS="CAT_DATA_DESC" FROM="(SELECT CD.CAT_DATA_DESC AS
CAT_DATA_DESC,C.PAR_KEY AS PAR_KEY FROM CS_DATOS_CLIENTE C INNER JOIN PHYSICAL_JOB P ON
C.PAR_KEY=P.PKEY_CUSTOMER LEFT JOIN CAT_DATA CD ON P.JOB_REASON_CODE=CD.CAT_DATA_CODE)
T "WHERE="CAT_DATA_DESC='Otros'" FILTERCOLUMN="PAR_KEY" FILTERPARAMETER="CUSTPKEY" #}
```

Ejemplo 3:

```
{# OPEN_CURSOR NAME="clientes" FROM="clientes as c" WHERE="c.edad
> 18"
ORDER_BY="c.nombre DESC" #}
```

Solo se procesarán los comandos OPEN_CURSOR cuyo cursor se utilice en el template.

-Para utilizar datos de un cursor en un template se utiliza la variable especial CURSOR_VALUE. La variable CURSOR_VALUE debe especificar los parámetros NAME y COLUMN de manera obligatoria. Opcionalmente pueden utilizarse los parámetros INDEX y SEPARATOR.

Por ejemplo (siguiendo con el ejemplo del cursor definido anteriormente):

```
{{ CURSOR_VALUE NAME="clientes" COLUMN="nombre" }}
```

Mostrará el nombre de todos los clientes de la tabla "clientes" que tengan más de 18 años, ordenado de forma descendente. La relación entre el CURSOR_VALUE y el cursor abierto anteriormente se hace a través del parámetro NAME.

-Si se especifica el parámetro index, el CURSOR_VALUE buscará la fila con ese índice en la DataTable de la variable con el nombre del cursor (ej: clientes) y para esa fila imprimirá el valor de la columna que dice el parámetro COLUMN. Ejemplo:

```
{{ CURSOR_VALUE name="clientes" COLUMN="nombre" index="0" }}
```

Mostrará el nombre de la primera fila que devuelva el query especificado al abrir el cursor.

-Si se especifica el parámetro separator, los resultados que devuelva el cursor se mostraran con el separador especificado, ejemplo:

```
{{ CURSOR_VALUE NAME="clientes" COLUMN="nombre" separator="," }}
```

Mostrará el nombre de todos los clientes de la tabla "clientes" que tengan más de 18 años, ordenado de forma descendente y separados por coma.

Ejemplo 1:

Template:

```
{# OPEN_CURSOR NAME="clientes" COLUMNS="NOMBRE,APELLIDO,FECHA_ALTA"  
FROM="CS_DATOS_CLIENTE as c" WHERE="c.MEDIO_PREFERIDO='EMAIL'" ORDERBY="c.nombre  
DESC" #}
```

Nombres clientes separados con espacio: {{ CURSOR_VALUE name="clientes" column="nombre" }}

Apellidos clientes separados con coma: {{ CURSOR_VALUE name="clientes" column="nombre" separator="," }}

Documento generado:

Nombres clientes separados con espacio: Pablo OSCAR JUAN Jose GERALDINE Esteban DANIEL
CYNTHIA ADRIAN

Apellidos clientes separados con coma: Pablo, OSCAR, JUAN, Jose, GERALDINE, Esteban, DANIEL,
CYNTHIA, ADRIAN

|

Como se puede observar en la imagen anterior, si no se especifica un separador, se utiliza el espacio como separador por defecto.

Ejemplo 2:

Template:

Prueba cursores

Acá de abre el cursor: {# OPEN_CURSOR NAME="clientes"
 COLUMNS="NOMBRE,APELLIDO,FECHA_ALTA,TIPO_PERSONA" FROM="CS_DATOS_CLIENTE as
 c" WHERE="c.MEDIO_PREFERIDO='EMAIL'" ORDERBY="c.nombre DESC" #}

Tabla de clientes

Nombre	Apellido	Fecha Alta	Tipo Persona
{{ CURSOR_VALUE name="clientes" column="NOMBR E" }}	{{ CURSOR_VALUE name="clientes" column="APELLID O" }}	{{ CURSOR_VALUE name="clientes" column="FECHA_ALT A" }}	{{ CURSOR_VALUE name="clientes" column="TIPO_PERSON A" }}
Ultima	Fila	De la	Tabla

Documento Generado:

Prueba cursores

Acá de abre el cursor:

Tabla de clientes

Nombre	Apellido	Fecha Alta	Tipo Persona
Pablo	Leston	19/06/2014 00:00:00	Juridica
OSCAR	SANTA CRUZ	14/10/2010 00:00:00	Fisica
JUAN	GOMEZ	04/02/2011 00:00:00	Fisica
Jose	Lopez	17/01/2017 14:34:48	Juridica
GERALDINE	BARDIN	09/06/2010 00:00:00	Fisica
Esteban	Gomez	14/12/2012 00:00:00	Fisica
DANIEL	TRIBLEHORN	13/07/2010 00:00:00	Juridica
CYNTHIA	BEREA	01/07/2008 00:00:00	Fisica
ADRIAN	SCARDILLO	08/06/2010 00:00:00	Fisica
Ultima	Fila	De la	Tabla

Como se puede observar en la imagen anterior, se respeta la tabla y estilos creados con el Word y se reemplazan los valores de cada columna haciendo uso de la variable CURSOR_VALUE.

Ejemplo 3:

Este ejemplo es de un reporte en el cual se puede observar que en el encabezado del Word se incluyen los títulos de las columnas para que en cada salto de página se vuelvan a mostrar. Luego se incluye una tabla y en cada columna se utiliza la variable cursor value haciendo referencia a la columna que se desea mostrar del query definido en el cursor abierto anteriormente en el template.

Template:

Nombre	Apellido	Fecha Alta	Tipo Persona
Encabezado <pre>{# OPEN_CURSOR NAME="clientes" COLUMNS="NOMBRE, APELLIDO, FECHA_ALTA, TIPO_PERSONA" FROM="CS_DATOS_CLIENTE" as c WHERE="c.MEDIO_PREFERIDO='EMAIL' OR DIRECCION='TUCUMAN 540, BUENOS AIRES'" ORDERBY="c.nombre DESC" #}</pre>			
{{ CURSOR_VALUE name="clientes" column="NOMBRE" }}	{{ CURSOR_VALUE name="clientes" column="APELLIDO" }}	{{ CURSOR_VALUE name="clientes" column="FECHA_ALTA" }}	{{ CURSOR_VALUE name="clientes" column="TIPO_PERSONA" }}
Ultima	Fila	De la	Tabla

Documento generado:

Nombre	Apellido	Fecha Alta	Tipo Persona
+			
ZULEMA LEONOR	CABA	08/04/1999 00:00:00	Fisica
SAMUEL	PEREZ	14/07/2001 00:00:00	Fisica
RUBEN ERNESTO	CAETE	27/01/2001 00:00:00	Fisica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica

PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
PruebaN	PruebaA		Juridica
Pablo	Leston	19/06/2014 00:00:00	Juridica
OSCAR	SANTA CRUZ	14/10/2010 00:00:00	Fisica
MIGUEL ANGEL	RABAIOTTI	17/11/2004 00:00:00	Fisica
MARTHA AMALIA	EBELING	18/04/2005 00:00:00	Juridica
MARIA SARA	SAADE	04/12/2004 00:00:00	Fisica
MARIA JOSEFA	MAAS		Fisica
MARANELA	SA MARTIN	11/08/2004 00:00:00	Fisica
ELIZABETH			

1

Nombre	Apellido	Fecha Alta	Tipo Persona
JUAN	GOMEZ	04/02/2011 00:00:00	Fisica
JOSEFINA	CAAMAÑO	04/03/2004 00:00:00	Fisica
ALEJANDRA			
JOSE BENITO	CAAMANO	31/07/1998 00:00:00	Fisica

Como se puede observar en la imagen anterior, al utilizar los títulos de las columnas en el encabezado del template, los mismos se mantienen al realizar saltos de página.

Generar Documento como adjunto

A partir de la **versión 5.9.0.1 Patch 6**, se agregó un nuevo parámetro "DocumentOutputDestination" al final del mensaje del servicio DOCX, con los valores FILE, PROCESS y CUSTOMER, para que el documento generado se pueda guardar en un archivo (como se venía haciendo hasta el momento) o se pueda guardar en la tabla PHYSICAL_ATTACHED_DOCUMENT (asociado al proceso o al cliente)

Mensaje de Entrada:

DOCX|CustPkey|JobPkey|UserId|CallTypeCode|DocxOutputFormat|**DocumentOutputDestination**[EOM]

DocumentOutputDestination = Valores Posibles: FILE, PROCESS, CUSTOMER

FILE: El documento generado se guarda en un archivo.

PROCESS: El documento generado se guarda en la tabla PHYSICAL_ATTACHED_DOCUMENT, asociado al proceso.

CUSTOMER: El documento generado se guarda en la tabla PHYSICAL_ATTACHED_DOCUMENT, asociado al cliente.

Ejemplos:

DOCX|0003916516_77|35f0d7ec-ede8-4385-9425-
f1cabab0cc8e|rolandob|ACT_TEST_DOCUMENT|DOCX[EOM]

DOCX|0003916516_77|35f0d7ec-ede8-4385-9425-
f1cabab0cc8e|rolandob|ACT_TEST_DOCUMENT|DOCX|FILE[EOM]

DOCX|0003916516_77|35f0d7ec-ede8-4385-9425-
f1cabab0cc8e|rolandob|ACT_TEST_DOCUMENT|DOCX|PROCESS[EOM]

DOCX|0003916516_77|35f0d7ec-ede8-4385-9425-
f1cabab0cc8e|rolandob|ACT_TEST_DOCUMENT|DOCX|CUSTOMER[EOM]