**Versión 1.6**

Compras, Pagos, ND,NC

**Versión 1.6.1**

Error en Frmpago y cobro, forma de pago. 04/02/2016

Se agregó grafico nuevo 04/02/2016

Se le agregó copiar y pegar correo electrónico en Frm clientes. 07/02/2016

El evento que llama myDataGrid1\_CellContentClick se quita del load, porque es el evento predeterminado o principal del objeto y por eso se ejecutaba dos veces. Pero era porque se llamaba en el load. Indicador ya no tiene uso. Esto es factura y compra, en Cobro ya se había quitado.07/02/2016

Se validó factura, compra, cobro y pago para que no permitan guardar documentos sin detalles.

Se permitió agregar anticipos por frm cobros y frm pagos, se habilitó la contabilización para estos documentos, ver empresa 994

En los formularios ajustes inventario, Ventas, Compras, pagos y cobros se muestran los registros ordenados por codcompracodigo, antes se ordenaban por fecha. Los registros no estabn ordenadose adecuadamente.Esto afectó las entidades de las mismas, en el método obtenertodos() y en la base de datos los metodods que tienen relación con paginación, se cambio fecha por codigo

Se modificaron los eventos de desplazamiento de paginas en ajustes inventario, Ventas, Compras, pagos y cobros, se leagregó dbBindingSource.MoveLast(); para que al cambiar d epagina se mueva al ultimo regiostro. Esto le da mejor percepción de las pagínas al usuario

**Version 1.6.2**

Se inició Modulo de caja y Bancos 15/02/2016

Se agregó Reinciar aplicación en el menú inicio 19/02/2016

Los formularios en el menú se desactivan de acuerdo a los permisos establecidos en el formulario de permisos. Se afectaron los formularios de permisos, acceso al sistema, la entidad permiso, formulario, y los módulos de MisRutinas. Tambien afecto la tabla de Formulario, se reclasificó. Seagregó un treeview en el formulario permiso. 19/02/2016.

Se corrigió error al colocar un código de articulo inexsitente en los detalles de factura de venta y compra, se afectó el evento txt\_validating. 20/02/2016

Se le agregó un campo a la lista desplegable de cuenta bancaria, en la parte forma de pago, tanto para Formulario Cobro y Pago, se afectó la entidad de negocio Cuenta Bancaria, Método BuscarTodos, y el evento EditingControl del grid afectado 21/12/2016

Cuando no hay conexión se le agregó un mensaje de direccionamiento a la ventana de configuración de conexiones., es mas amigable que los mesnajes de errores. Se creó un método en la capa de datos, en la entidad datos de empresa y en el load del acceso para hacer la prueba de conexión y direccionar 21/02/2016

**1.6.3**

En el formulario Articulo, no se podrá actualizar el stock, en el formulario caja no se podrá actualizar el saldo, ni tampoco en cuenta bancaria, porque dependen de movimientos., si se dejase que el usuario modifique estos valores, podríamos tener valores no confiables, ya que si alguien abre el formulario de artículos, y en ese momento en esa u otra estación de trabajo se hacen movimientos como ventas, compras o ventas se actualizará en la base de datos en la tabla articulo el stock y será un valor diferente al que está formulario abierto, y si el usuario guarda en el formulario abierto se perdería el valor real por un valor ya obsoleto.

Corregido error en validación de factura de venta y compra. Donde se valida que la factura no quede sin registros.

**1.7**

En todos los procedimientos almacenados(BD) de paginado

Como por ejemplo dbo.spDocumccPaginado, se hizo la modificación de colocarle el where dentro de la consulta base. 01/03/2016

.

.

.

SELECT tipodoc,numero,status,codcliente,fecemis,fecvenc,codvendedor,tipodocaso,numerodocaso,numerocontrol,numerocomp,descripcion,codimp,totalexento,totalimponible,porcimp,totalimpuesto,totalneto, saldo,numeroplanillaexp

FROM ( SELECT Documcc.\*,

(row\_number() over( PARTITION by codempresa order by fecemis)) as rownumber

FROM Documcc WHERE codempresa=@codempresa

.

.

.

En el proyecto Aplicacion Contable, en la carpeta Utilidades, Variables Globales, se cambió la constante local Glo\_CantidadRegistroPagina a 100, que alimenta la propiedad pública CantidadRegistroPagina que representa la cantidad de registros que se muestran por página, en los formularios de movimientos donde se usa paginación. Esto hace mas ligero el sistema con el uso de bases de datos remotas. 01/03/2016

Validación en guardar, formulario de Cobros y Pagos. Que el cliente coincida con el nro de movimiento y también que los movimientos coincidan con el cliente. 02/03/2016

Al formulario frmcobro se le agregó el campo codvendedor.02/03/2016

Se le agrega la validación en activar y anular cobros y pagos esto es en los sp de la BD. Validacion en caso de que sean solo facturas manuales

10/03/2016

-- y si la facura no es manual, si existe en factura

if ((select count(\*) from Factura where codfactura=@numero and codempresa=@codempresa)>0)

Se le agrega a la tabla comprobante el campo descripcióngeneral, y por ello a la entidad en al capa de negocios y al formulario 14/03/2016

Se Desactiva la validación en docum cxc y docum cxp “Documento no Existe”, ya que no está completa. Es temporal, 19/03/2016

Correción en el botón Pagar todos, en la línea

Documcc doc = new Documcc(myDataGrid1.Rows[i].Cells["gstxttipodoc"].Value.ToString(), Int32.Parse(myDataGrid1.Rows[i].Cells["gstxtnumero"].Value.ToString()), VariablesGlobales.CodEmpresa);

Por

Documcp doc = new Documcp(myDataGrid1.Rows[i].Cells["gstxttipodoc"].Value.ToString(), Int32.Parse(myDataGrid1.Rows[i].Cells["gstxtnumero"].Value.ToString()), VariablesGlobales.CodEmpresa);

Corrección en los procedimientos almacenados de estado de cuenta de clientes y proveedores. 20/03/2016

Se incorporó un menú estilo Modern UI. Finalizado el 20/03/2016

Validación en formulario de permisos, al guardar, se valida que el perfil se haya seleccionado, que no este en blanco

Finalizado el modulo de caja y bancos. Gracias a Dios 22/03/2016

**1.7.1**

Se agrega el proyecto Herramientas 1.0 para Contable.Net 1.7.1, donde se puede hacer mantenimiento a las tablas de países, estados ciudades, estructura contable, configuración de reportes, y otros. 10/04/2016.

En Documentos de cxc y cxp se agrega la validación para que no permita editar los registros provenientes de cobros, pagos, ni de factura de venta y compra.

Se corrige la validación al anular cobros y pagos que tienen un registro relacionado en pago de tipo Adelanto. Si ese documento(ADEL) que se pretende anular tiene saldo diferente del total neto, es decir, si ya fue cruzado, no se podrá anular. 10/04/2016.

Al anular o reactivar un cobro o pago, si tiene un registro relacionado en documentos (Adelanto), lo anula o reactiva también, 11/04/2016

Al formulario Documentos de cxc y cxp se le agregó anular y reactivar, solo documentos que no provengan de otros módulos ( de cobro, pago, factura venta, y compra) 12/04/2016

Los frames de búsquedas de los formularios de movimientos ahora se pueden mover en la pantalla. 12/04/2016

Se cambió la forma de hacer las validaciones del campo Status o anulado, ya que antes se hacía consultando el objeto en el formulario(txtstatus) y ahora se valida haciendo solicitudes a la base de datos. Se hace para evitar errores 14/04/2016

Cuando el sistema Contable.Net se Conectaba a la base de datos en la nube, el procedimiento(PermisosMenu) que se encuentra en la Clase MisRutinas, llamado desde el formulario Acceso, hacía un recorrido por el menú y por cada opción hacía una solicitud en la BD, entonces se tornaba lento el acceso, por lo tanto se procedió a solicitar los datos requeridos(permisos y formularios) en datatables y así hacer validaciones de manera local, el rendimiento es notable.15/04/2016

**HERRAMIENTA para Contable 1.7.1 pasa de la V 1.0.0 a la V 1.0.1**

Se actualiza, ya que la versión 1.0.0 no permitía conexiones dinámicas a las Base de Datos 13/05/2016

**Contable 1.7.2**

Corrección en el procedimiento Contabilizar Documentos, cuando se contabiliza Notas de Créditos cxc con Detalles, se estaba seleccionado la variable @cuentaingreso, cuando sabemos que en la nota de crédito a clientes se usa una cuenta de egreso que justifica una devolución. Esta corrección también se hace sobre el asiento contable de ejemplo en la empresa 944 21/05/2016

Este error afecta la BD, por tanto revisar el Script de actualización 1.7.2, buscar como /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Actualización: 1\*/

De igual manera aplica para el procedimiento [spActualizarInventarioDeleted] Buscar como:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Actualización: 3\*/

En el procedimiento [dbo].[spActualizarInventarioInserted]

La longitud del tamaño del tipo de datos de la variable @codarticulo es de 10, cuando el tamaño máximo es de 50 según la tabla. Esto ocasionaba problemas a la hora de usar códigos de artículos con longitud mayor a 10. Entonces se debe modificar a 50

ALTER PROCEDURE [dbo].[spActualizarInventarioInserted]

@codarticulo varchar(50), 21/05/2016 Este error afecta la BD, por tanto revisar el Script de actualización 1.7.2, buscar como /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Actualización: 2\*/

En los formularios Empresa y Artículos se cambia el uso de la propiedad BackgroundImage por Image, que es la correcta. También se agrega un formulario para darle una vista previa a la imagen. No afecta a la base de datos. 22/05/2016

En los formularios de cobros y pagos se corrige el procedimiento validarsinregistros, el cual arrojaba el mensaje sobre la vaidación donde los detalles no podían quedar sin registros. 01/06/2016

Se Corrige en el botón buscar de los formularios Documentos de Cuentas por cobrar y Documentos de cuentas por Pagar, la validación del campo wtxttipodoc 04/06/2016

Tipodoccc tdo = new Tipodoccc(this.wtxttipodoc.Text);

if (this.wtxttipodoc.Text.Trim() != "" && tdo.ExisteReg(this.wtxttipodoc.Text) == false)

{

MessageBox.Show("El código seleccionado no existe en la base de datos en Tipo de Documentos", "Sistemas Administrativos", MessageBoxButtons.OK, MessageBoxIcon.Information);

wtxttipodoc.Focus();

SendKeys.Send("{HOME}+{END}");

return;

}

En el procedimiento almacenado RptExistenciaAlmacenesHorizontal se le agrega la condición where codempresa=@codempresa a la consulta que selecciona los almacenes existentes.

set nocount on

SELECT ROW\_NUMBER() OVER( ORDER BY codalmacen ASC ) AS fila, codalmacen

into #temp1

FROM almacen where codempresa=@codempresa

13/06/2016

**En el Formulario de Comprobantes se agrega un botón en el detalle para explorar las cuentas contables, y asi el usuario pueda hacer una búsqueda de ellas. También se agrega un botón debajo de los detalles para que el usuario pueda actualizar las listas solamente de los detalles. 15/06/2016**

Corrección en el reporte [sprptLibroMayorDet] Libro mayor, en el where:

where c.numerocomprobante=dc.numerocomprobante and c.mes=dc.mes and c.año=dc.año and c.ejercicio=dc.ejercicio and c.codempresa=dc.codempresa and

dc.codempresa=@codempresa and dc.ejercicio=@ejercicio and dc.año=@año and dc.mes=@mes

18/06/2016

El tamaño de las variables @strsql y @ssss en el procedimiento spMovimientoBancoPaginadobusqueda fue cambiado a 2000. Cuando se hacen consultas largas usando estas variables se truncaba la sentencia. Esto también puede implicar en el resto de las sentencias de paginado. Se aplicó también a spMovimientoCajaPaginadobusqueda 19/06/2016

**Se agregan 3 métodos a los formularios de mantenimiento de tablas maestros y movimientos. Búsqueda rápida por los campos principales, actualización de todos las listas y actualización de todos los datos incluyendo las listas.19/06/2016**

Se mejoró validación en el evento SelectedIndexChanged del objeto cmbiva del formulario Orden de Pago

private void cmbiva\_SelectedIndexChanged(object sender, System.EventArgs e)

{

if (Equivalencias.IsNumeric(txtmontomenosretencion.Text) == true && cmbiva.SelectedValue!=null)

{

Impuesto imp = new Impuesto();

double tasa = imp.ObtenerTasa(dtpfecha.Value, cmbiva.SelectedValue.ToString());

double total = tasa \* double.Parse(txtmontomenosretencion.Text);

txtmontoiva.Text = String.Format("{0:#,0.00}", total);

}

}

20/06/2016

**Contable VB.Net 1.7.2**

Se convierte el sistema a VB.NET

**Contable C# 1.8.0**

Se cambia validación en anular, cobrar, pagar y eliminar factura de compra y venta. Cuando se acude a el txtstatus para validar el status. Ahora se acude a la base de datos llamando la clase para hacer referencia al status. 15/07/2016

En las entidades de Compra y Factura, en los métodos Modificar se quitó el campo status. No debe estar allí, causará que se borre el status en las tablas cuando se modifique un registro. 16/07/2016

Se depuró una la trigger update en la tabla detallecompra, no coincidían los campos en el into cursor 16/07/2016

Se le agregó Pedidos al Módulo de Cuentas por cobrar, con la funcionalidad que se puede importar desde la factura, maneja pendientes y actualiza stock comprometido. Se afectaron las bases de datos(Archivo Sql: Actualizaciones 1.8.0),también los formularios de factura, empresas. 20/07/2016

Se le agrega un botón en el menú principal para recargar el formulario actual. 23/07/2016

**ACTUALIZACION DE BASE DE DATOS 1.8.0 19/08/2016**

Se corrige el procedimiento SpCrearEmpresa en la instrucciones:

if (@copiarplan=1)

if (@copiarcentro=1)

if (@copiartipo=1)

Quedando de la siguiente manera

if (@copiarplan=1 and (select count(\*) from PlanContable where

codempresa=@codempresaorigen)>0 )

if (@copiarcentro=1 and (select count(\*) from CentroCostos where

codempresa=@codempresaorigen)>0

)

if (@copiartipo=1 and (select count(\*) from tipocomprobante where

codempresa=@codempresaorigen)>0)

Al crear una empresa por el asistente, al copiar estas tablas de una empresa a otra, y no tiene registros para copiar la triger de validación de la tabla enviará un mensaje de error, entonces evitamos este mensaje **19/08/2016 EJECUTAR SCRIPT DE ACTUALIZACION.**

**ACTUALIZACION DE BASE DE DATOS 1.8.0 19/08/2016**

Se le agregó a estos procedimientos la cláusula set dateformat dmy. El formato de fechas de servidores en la nube pueden variar de nuestros servidores locales, por lo que se le asigna el formato a las fechas cuando estamos uniendo cadenas para crear fechas.

RptFlujoDeCaja

spcrearempresa

RptChequeEstandar

Para detectar los procedimiento se usó

select name, text

from sysobjects c\_obj left join syscomments com on c\_obj.id = com.id

where (c\_obj.xtype = 'P' or c\_obj.xtype = 'TR') and (text like '%convert(date%' or text like '%convert(smalldate%')

**19/08/2016 EJECUTAR SCRIPT DE ACTUALIZACION.**

**ACTUALIZACION DE BASE DE DATOS 1.8.1 22/08/2016**

Se quita el [Contable].[dbo] en los siguientes procedimientos de la base de datos.

Esto arrojará un error cuando la base de datos no se llame Contable

spActualizarInventarioInserted

spCierreContableEjercicio

spActualizarInventarioDeleted

spActualizarInventarioDeletedComprometido

spActualizarInventarioInsertedComprometido

Para localizarlos se usó:

SELECT

so.name,

sc.text

FROM

sysobjects so

INNER JOIN syscomments sc ON so.id = sc.id

WHERE

(so.type = 'P' so.type = 'FN' or so.type = 'TR') AND

sc.text LIKE '%![Contable!].![dbo!].%' ESCAPE '!'

23/08/2016 Se corrige una validación en el formulario Documentos por Pagar, en el botón Anular, evento clic. Se cambió Documcc doc4 = new Documcc por Documcp doc4 = new Documcp

Esta es bloque de código afectado:

if (txttipodoc.Text.Trim() != "" && Int32.TryParse(this.txtnumero.Text.Trim(), out s) == true)

if (docc.ExisteReg(this.txttipodoc.Text, Int32.Parse(txtnumero.Text), VariablesGlobales.CodEmpresa) == true)

{

Documcp doc4 = new Documcp(this.txttipodoc.Text, Int32.Parse(txtnumero.Text), VariablesGlobales.CodEmpresa);

Compra fac = new Compra();

if (doc4.tipodoc == "FACT")

if (fac.ExisteReg(doc4.numero, VariablesGlobales.CodEmpresa) == true)

{

MessageBox.Show("El documento no se puede anular, proviene de FACTURA de Compra", "Sistemas Administrativos", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtnumerodocaso.Focus();

SendKeys.Send("{HOME}+{END}");

return;

}

}

**1.8.2 Contable.Net**

Se Agrega cálculo de retenciones, métodos que se encuentran en Factura, Compra, Documentos cxc, Documentos cxp.

Se usan los campos Porcentaje de retención en Clientes y Proveedores.

Se agregan los procedimientos almacenados spCrearRetencioncxc, spCrearRetencioncx, spcorreDOCCXCRETURN, spcorreDOCCXPRETURN 29/08/2016

**1.8.2 Contable.Net**

Formulario Documentos cxp, en el boton guarder se corrige la validación siguiente:

Documcc doc4 = new Documcc por Documcp doc4 = new Documcp 01/09/2016

if (txttipodoc.Text.Trim() != "" && Int32.TryParse(this.txtnumero.Text.Trim(), out s) == true)

if (docc.ExisteReg(this.txttipodoc.Text, Int32.Parse(txtnumero.Text), VariablesGlobales.CodEmpresa) == true)

{

Documcp doc4 = new Documcp(this.txttipodoc.Text, Int32.Parse(txtnumero.Text), VariablesGlobales.CodEmpresa);

Compra fac = new Compra();

if (doc4.tipodoc == "FACT")

if (fac.ExisteReg(doc4.numero, VariablesGlobales.CodEmpresa) == true)

{

MessageBox.Show("El documento no se puede modificar, proviene de FACTURA de Compra", "Sistemas Administrativos", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtnumerodocaso.Focus();

SendKeys.Send("{HOME}+{END}");

return;

}

}

**1.8.2 Contable.Net 05/09/2016**

En el formulario de Plan Contable, en tipos de Patrimonio, La cuarta opción está repetida, se cambió por “Otras Cuentas de Patrimonio”

**1.8.2 Contable.Net 05/09/2016**

En el evento CellContentClick de todos los grid del formulario ParametrosContables agregar:

busc.Quien = "FrmParametrosContables";

antes de

busc.ShowDialog();

En el formulario BuscarCuentaContable en el botón y evento

private void cmdseleccionar\_Click(object sender, EventArgs e)

Cambiar if (Quien == "") por

if (Quien == "FrmParametrosContables")

**1.8.2 Contable.Net 09/09/2016**

Corrección procedimiento almacenado [RptEstadoCuentaProveedor]

Ver archivo de actualziaciones

FROM [dbo].Pago, DetallePagoForma d

where Pago.codproveedor=@codproveedor and (fecha between @fechaini and @fechafin) and pago.codempresa=@codempresa and d.codempresa=@codempresa

and anulado=0 and monto <>0 and pago.codpago=d.codpago and pago.codempresa **=d.codpago**

POR

FROM [dbo].Pago, DetallePagoForma d

where Pago.codproveedor=@codproveedor and (fecha between @fechaini and @fechafin) and pago.codempresa=@codempresa and d.codempresa=@codempresa

and anulado=0 and monto <>0 and pago.montototal<>0 and pago.codpago=d.codpago and pago.codempresa =

**=d.codempresa**

Y en el procedimiento almacenado [RptEstadoCuentaCliente]

Ver archivo de actualizaciones

FROM [dbo].Cobro , DetalleCobroForma d

where cobro.codcliente=@codcliente and (fecha between @fechaini and @fechafin) and cobro.codempresa=@codempresa

and anulado=0 and d.monto <>0 and cobro.codcobro=d.codcobro and cobro.codempresa=d.codempresa

Por

FROM [dbo].Cobro , DetalleCobroForma d

where cobro.codcliente=@codcliente and (fecha between @fechaini and @fechafin) and cobro.codempresa=@codempresa

and anulado=0 and d.monto <>0 and cobro.montototal<>0 and cobro.codcobro=d.codcobro and cobro.codempresa=d.codempresa

**1.8.2 Contable.Net 10/09/2016**

Formulario Estado de Cuentas de Proveedores, Corregido el título del mismo.

**1.8.2 Contable.Net 10/09/2016**

En el Formulario Factura de Compras, en el método aceptar, se usa una función llamada validarsinregistros, buscarla y corregir:

// valida monto abono

if (myDataGrid1.Rows[i].Cells[4].Value == null)

cond2 = 0;

else

if (double.Parse(myDataGrid1.Rows[i].Cells[4].Value.ToString()) <= 0)

cond2 = 0;

Por

// valida monto abono

if (myDataGrid1.Rows[i].Cells[5].Value == null)

cond2 = 0;

else

if (double.Parse(myDataGrid1.Rows[i].Cells[5].Value.ToString()) <= 0)

cond2 = 0;

**1.8.2 Contable.Net 10/09/2016**

En la TRIGGER [dbo].[InsDetalleDocumcp] de la tabla DetalleDocumcp se corrije:

if isnull((select count(\*) from inserted i, **documcc** t

where i.tipodoc= t.tipodoc and i.numero=t.numero and i.codempresa=t.codempresa),0)<>@totalregistrosafectados

begin

RAISERROR

(N'Nro de Documento no existe. Uno o mas registros no están relacionado con la tabla de Documento',10,1)

if @@trancount>0

begin

rollback transaction;

RETURN;

end

end

Por

if isnull((select count(\*) from inserted i, **documcp** t

where i.tipodoc= t.tipodoc and i.numero=t.numero and i.codempresa=t.codempresa),0)<>@totalregistrosafectados

begin

RAISERROR

(N'Nro de Documento no existe. Uno o mas registros no están relacionado con la tabla de Documento',10,1)

if @@trancount>0

begin

rollback transaction;

RETURN;

end

end

**1.8.2 Contable.Net 10/09/2016**

En la TRIGGER [dbo].[UpdDetalleDocumcp] de la tabla DetalleDocumcp

Se corrije:

if isnull((select count(\*) from inserted i, **Documcc** t

where i.tipodoc=t.tipodoc and i.numero=t.numero and i.codempresa=t.codempresa),0)<>@totalregistrosafectados

begin

RAISERROR

(N'Nro de Documento no existe. Uno o mas registros no están relacionado con la tabla de Documentos',10,1)

if @@trancount>0

begin

rollback transaction;

RETURN;

end

end

Por

if isnull((select count(\*) from inserted i, **Documcp** t

where i.tipodoc=t.tipodoc and i.numero=t.numero and i.codempresa=t.codempresa),0)<>@totalregistrosafectados

begin

RAISERROR

(N'Nro de Documento no existe. Uno o mas registros no están relacionado con la tabla de Documentos',10,1)

if @@trancount>0

begin

rollback transaction;

RETURN;

end

end

**1.8.2 Contable.Net 11/09/2016**

Se creó una nueva tabla, llamada ImpuestoParametro, en el que se guardan los valores de impuesto general y exento. Estos valores pueden ser actualizados por el formulario Tipo de Impuestos.

Se hicieron modificaciones en:

Función FCodigoImpuestoExento en sql server

Función FCodigoImpuestoGEneral en sql server

Entidad TipoImpuesto en capa de negocio

Formulario FrmTipoImpuesto en capa de interfaces.

**1.8.2 Contable.Net 15/09/2016**

El tipo de datos de los campos stockcomprometido de las tablas Articulo y Stalma deben ser cambiados de int a decimal(18,4). Ver script de actualización en la parte:

GO

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*reparando stockcomprometido de int a decimal. No se pierden los valores

--elimina la restriccion

ALTER TABLE [dbo].[Articulo] DROP CONSTRAINT [DF\_\_Articulo\_\_stockc\_\_66C02818]

--cambia el tipo de datos

alter table articulo alter column stockcomprometido decimal(18,4) not null

--agrega la restricccion con el nombre que tenia anteriormente

ALTER TABLE [dbo].[Articulo] ADD CONSTRAINT [DF\_\_Articulo\_\_stockc\_\_66C02818] DEFAULT ((0)) FOR [stockcomprometido]

GO

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*reparando stockcomprometido de int a decimal. No se pierden los valores

--elimina la restriccion

ALTER TABLE [dbo].STAlma DROP CONSTRAINT [DF\_\_STAlma\_\_stockcom\_\_65CC03DF]

--cambia el tipo de datos

alter table STAlma alter column stockcomprometido decimal(18,4) not null

--agrega la restricccion con el nombre que tenia anteriormente

ALTER TABLE [dbo].STAlma ADD CONSTRAINT [DF\_\_STAlma\_\_stockcom\_\_65CC03DF] DEFAULT ((0)) FOR [stockcomprometido]

GO

**1.8.2 Contable.Net 21/09/2016**

En el Formulario de Factura de Venta, en el botón de “Cobrar”, se agregó y modificó el siguiente código, que busca el primer registro perteneciente a una caja, que luego se envía al formulario de Cobros en el grid de forma de pago. Y la instrucción fa.fecemis.ToString("dd/MM/yyyy") que envía la fecha del registro y evita un mensaje de error de tipo fecha incorrecto porque se enviaba vacío

Caja caj = new Caja(VariablesGlobales.CodEmpresa);

if (caj.CantReg(VariablesGlobales.CodEmpresa) == 0)

{

MessageBox.Show("No existen cajas creadas. Cree al menos una caja para usar esta opción");

return;

}

string[] filagrid2 = { "EFEC", "", "", fa.fecemis.ToString("dd/MM/yyyy"), "\*", "0", caj.codcaja, "N/A", String.Format("{0:#,0.00}", variabletotal) };

**1.8.2 Contable.Net 21/09/2016**

En el Formulario de Compra, en el botón de “Pagar”, se agregó y modificó el siguiente código, que busca el primer registro perteneciente a una caja, que luego se envía al formulario de Pagos en el grid de forma de pago. Y la instrucción fa.fecemis.ToString("dd/MM/yyyy") que envía la fecha del registro y evita un mensaje de error de tipo fecha incorrecto porque se enviaba vacío

Caja caj = new Caja(VariablesGlobales.CodEmpresa);

if (caj.CantReg(VariablesGlobales.CodEmpresa) == 0)

{

MessageBox.Show("No existen cajas creadas. Cree al menos una caja para usar esta opción");

return;

}

string[] filagrid2 = { "EFEC", "", "", fa.fecemis.ToString("dd/MM/yyyy"), "\*", "0", caj.codcaja, "N/A", String.Format("{0:#,0.00}", variabletotal) };

**1.8.2 Contable.Net 25/09/2016**

Se agrega Despacho en el módulo de Cuentas Por cobrar, y varios reportes.