

Customers (part of the **CMS.Ecommerce** namespace) and users (part of the **CMS.Membership** namespace) are different entities within Kentico. To avoid unnecessary configuration, the system contains mechanisms for synchronizing the properties of users and customers.

The synchronization uses methods in the **CustomerInfoProvider** class in the **CMS.Ecommerce** namespace and runs in two ways:

- [From customer properties to user properties](#)
- [From user properties to customer properties](#)

From customer properties to user properties

If a customer's properties are changed, the **CopyDataFromCustomerToUser** method is triggered.

Properties synchronized by default

	Column name in the database	
	From	To
First name	COM_Customer.CustomerFirstName	CMS_User.FirstName
Last name	COM_Customer.CustomerLastName	CMS_User.LastName
Full name	COM_Customer.CustomerFirstName + CMS_User.MiddleName + COM_Customer.CustomerLastName	CMS_User.FullName
Email	COM_Customer.CustomerEmail	CMS_User.Email
Phone	COM_Customer.CustomerPhone	CMS_UserSettings.UserPhone

Overriding the default synchronization

To change the default synchronization, override the **CopyDataFromCustomerToUserInternal(CustomerInfo customer, UserInfo user)** method.



```
protected virtual bool CopyDataFromCustomerToUserInternal(CustomerInfo customer,
UserInfo user)
{
    bool updated = false;

    // Checks that both customer and user are not null
    if ((customer == null) || (user == null))
    {
        return false;
    }

    // Checks whether the customer's first name was updated
    if (!string.Equals(customer.CustomerFirstName, user.FirstName))
    {
        user.FirstName = customer.CustomerFirstName;
        updated = true;
    }

    // Checks whether the customer's email was updated
    if (!string.Equals(customer.CustomerEmail, user.Email))
    {
        user.Email = customer.CustomerEmail;
        updated = true;
    }

    // Checks whether the customer's last name was updated
    if (!string.Equals(customer.CustomerLastName, user.LastName))
    {
        user.LastName = customer.CustomerLastName;
        updated = true;
    }

    // Checks whether the customer's phone number was updated
    if (!string.Equals(customer.CustomerPhone, ValidationHelper.GetString(user.
UserSettings.GetValue("UserPhone"), "")))
    {
        user.UserSettings.SetValue("UserPhone", customer.CustomerPhone);
        updated = true;
    }

    // Sets the full name from the first, middle and last names
    user.FullName = UserInfoProvider.GetFullName(customer.CustomerFirstName, user.
MiddleName, customer.CustomerLastName);

    // Returns a bool value that indicates whether any of the customer's
synchronized properties were changed
    return updated;
}
```

From user properties to customer properties

If a user's properties are changed, the **CopyDataFromUserToCustomer** method is triggered.

Properties synchronized by default

Column name in the database

	From	To
First name	CMS_User.FirstName	COM_Customer.CustomerFirstName
Last name	CMS_User.LastName	COM_Customer.CustomerLastName
Email	CMS_User.Email	COM_Customer.CustomerEmail
Phone	CMS_UserSettings.UserPhone	COM_Customer.CustomerPhone

Overriding the default synchronization

To change the default synchronization, override the **CopyDataFromUserToCustomerInternal(CustomerInfo customer, UserInfo user)** method.



```
protected virtual bool CopyDataFromUserToCustomerInternal(CustomerInfo customer,
UserInfo user)
{
    bool updated = false;

    // Checks that both customer and user are not null
    if ((customer == null) || (user == null))
    {
        return false;
    }

    // Checks whether the user's first name was updated
    if (!string.Equals(customer.CustomerFirstName, user.FirstName))
    {
        customer.CustomerFirstName = user.FirstName;
        updated = true;
    }

    // Checks whether the user's email was updated
    if (!string.Equals(customer.CustomerEmail, user.Email))
    {
        customer.CustomerEmail = user.Email;
        updated = true;
    }

    // Checks whether the user's last name was updated
    if (!string.Equals(customer.CustomerLastName, user.LastName))
    {
        customer.CustomerLastName = user.LastName;
        updated = true;
    }

    // Checks whether the user's phone was updated
    if (!string.Equals(customer.CustomerPhone, ValidationHelper.GetString(user.
UserSettings.GetValue("UserPhone"), "")))
    {
        customer.CustomerPhone = ValidationHelper.GetString(user.UserSettings
["UserPhone"], "");
        updated = true;
    }

    // Returns a bool value that indicates whether any of the user's synchronized
properties were changed
    return updated;
}
```