

# Гаджеты для боковой панели Windows Vista

## Краткое руководство

© Никита Культин  
<http://kultin.ru>

### Размещение

Стандартные гаджеты Windows Vista (**Календарь**, **Контакты**, **Погода** и др.) находятся в папке:  
C:\Program Files\Windows Sidebar\Gadgets

Гаджеты, *установленные пользователем*, находятся в папке:  
C:\Users\**UserName**\AppData\Local\Microsoft\Windows Sidebar\Gadgets

### Создание гаджета

1. Создать папку: *aGadget*, где *aGadget* - имя гаджета
2. В созданной папке создать:
  - папку CSS – для css-файлов;
  - папку JS – для js-файлов;
  - drag.png – картинка, которая будет изображать гаджет в процессе его перемещения (перетаскивания) из галереи на боковую панель Windows Vista;
  - icon.png - картинка (48x48), которая будет изображать гаджет в галереи гаджетов;
  - logo.png - логотип, отображается на странице галереи гаджетов;
  - aGadget.html - главная (стартовая) страница гаджета;
  - gadget.xml - манифест, содержит информацию о гаджете и ссылку на html-файл
3. Содержимое папки *aGadget* упаковать в zip-архив *aGadget.zip*. Затем изменить расширение файла архива с zip на gadget.
4. Открыть папку, в которой находится файл *aGadget.gadget*, и сделать двойной щелчок на значке файла. В результате будет активизирован процесс установки гаджета, по завершении которого в галереи гаджетов появится значок гаджета.
5. Чтобы активизировать гаджет, надо сделать двойной щелчок на значке гаджета или перетащить значок на боковую панель.

### Фоновый рисунок

Стандартный размер окна гаджета (фоновый рисунок) – 130x150.  
Создать фоновый рисунок png-формата с прозрачным фоном можно в Adobe Illustrator.  
Основой для рисунка является прямоугольник.

1. Поместить на рабочий лист прямоугольник 122x142
2. В меню Effect выбрать команду Stylize>>Drop Shadow
3. Задать параметры тени: X Offset: 1; Y Offset: 1; Blur:1.
4. Сохранить рабочий лист в png-формате: File >>Export и т.д.

В результате в созданном файле будет находится картинка размера 130x150.

Имя фоновой картинке следует указать в качестве параметра background тега body:

```
<body unselectable="on" scroll="yes" onload="loadMain()" background = "url(images/u-7.png)">
```

Также надо задать значения параметров width и height в определении стиля body. В качестве значения параметров следует указать, соответственно, ширину и высоту картинке:

```
body  
{  
  width: 130px;  
  height: 150px;
```

```
font-size: 8pt;
font-family: Tahoma;
cursor: default;
}
```

## Манифест

```
gadget.xml
<?xml version="1.0" encoding="utf-8"?>
<gadget>
  <name>Название</name>
  <namespace> microsoft.windows</namespace>
  <version>1.0.0.0</version>
  <author name="Культин Н.Б.">
    <info url="http://www.kultin.ru"/>
    <logo src="logo.png"/>
  </author>
  <copyright>© 2007</copyright>
  <description>Краткое описание</description>
  <icons>
    <icon height="48" width="48" src="icon.png"/>
  </icons>
  <hosts>
    <host name="sidebar">
      <base type="html" apiVersion="1.0.0" src="tasks.html"/>
      <permissions>full</permissions>
      <platform minPlatformVersion="1.0"/>
      <defaultImage src="drag.png"/>
    </host>
  </hosts>
</gadget>
```

## Доступ к переменным среды

Переменные среды:

APPDATA - C:\Users\UserName\AppData\Roaming

HOME\PATH - \Users\UserName

TEMP и TMP - C:\Users\UserName\AppData\Local\Temp

USERNAME - имя пользователя

Пример:

```
var result = document.getElementById("result");
result.innerHTML = SystemEnvironment.Variable("APPDATA");
```

## Объект System.Gadget

Событие	Описание
onDocked	
onSettingsClosed	
onSettingsClosing	
onShowSettings	
onUndock	

Метод	Описание
beginTransition	
close	
endTransition	

Свойство	Описание
background	фоновый рисунок
docked	
name	имя гаджета, указанное в манифесте
opacity	степень непрозрачности (целое)
path	путь к каталогу гаджета
platformVersion	
settingsUI	
version	

## Среда разработки

В качестве среды разработки можно использовать редактор сценариев Microsoft Word.

Чтобы приступить к созданию главной страницы гаджета надо:

- В меню **Сервис** выбрать команду **Макрос ▶ Редактор сценариев**
- В меню **File** открывшегося окна редактора сценариев выбрать команду **New ▶ File**
- В окне **New File** выбрать **HTML Page** и сделать щелчок на кнопке **Open**

Если главная страница уже существует, то для того чтобы загрузить ее в редактор сценариев надо:

- Открыть окно редактора сценариев (команда **Сервис ▶ Макрос ▶ Редактор сценариев**)
- В меню **File** открывшегося окна редактора сценариев выбрать команду **Open ▶ File**

Другие страницы гаджета (например, страница параметров) создается и открывается аналогично.

После того как страница будет создана, надо задать язык сценария страницы (VB Script или Java Script) и метод размещения элементов управления.

Чтобы задать язык сценария страницы надо:

- В меню **View** выбрать команду **Property Pages**
- Открыть вкладку **General**
- В группе **Default script language** раскрыть список **Client** и выбрать **VB Script** или **Java Script**

Чтобы задать метод размещения элементов управления на странице надо:

- В меню **View** выбрать команду **Property Pages**
- Открыть вкладку **General**
- В списке **Page Layout** выбрать **GridLayout**

## Страница Параметры

Страница **Параметры** появляется в результате щелчка на кнопке **Параметры**.

Для того чтобы кнопка **Параметры** стала доступной (появилась), в текст функции обработки события Load главной страницы гаджета надо добавить инструкцию

System.Gadget.settingsUI = "Settings.html";  
где: Settings.html – страница параметров.

Пример

### Листинг . main.html

---

```
<html>
```

```
<head>
```

```

<title>Gadget name</title>

<link href="css/nk.css" type="text/css" rel="stylesheet" />
<script src="js/nk.js" language="javascript" type="text/javascript"></script>

<script>
    // обработка событий
    //System.Gadget.onShowSettings = ShowSettings;    // отображение страницы параметров
    System.Gadget.onSettingsClosed = SettingsClosed; // страница параметров закрыта
</script>

</head>

<body unselectable="on" scroll="no" onload="loadMain()" BACKGROUND="url(images/background.png)">

<span>
    <b>nkGadget</b>
    <div id="info"></div>
</span>

</body>

</html>

```

### Листинг . nk.js (фрагмент)

---

```

function loadMain()
{
    System.Gadget.settingsUI = "Settings.html";
}

```

Листинг: APIs-Gadgets.html

```

<html>
<head>
    <title>API Gadget Demo</title>
    <meta http-equiv ="Content-Type" content="text/html; charset=utf-8">
    <style>
        body
        {
            margin: 0;
            width: 230px;
            height: 260px;
        }
        #gadgetContent
        {
            width: 230px;
            height: 260px;
            top: 0px;
            text-align: center;
            font-family: Tahoma;
            font-size: 10 pt;
            position: absolute;
        }
    </style>

```

```

</style>
<script>
    // обработка событий
    System.Gadget.onUndock = dockedState;
    System.Gadget.onDock = dockedState;
    System.Gadget.onSettingsClosed = SettingsClosed;
    System.Gadget.onShowSettings = ShowSettings;

    // the gadget is docked/undocked
    function dockedState()
    {
        DisplayValue("Docked: " + System.Gadget.docked);
    }

    function ShowSettings(event)
    {
        DisplayValue("Settings page opened");
    }

    function SettingsClosed(event)
    {
        DisplayValue("Settings Closed: " + event.closeAction);
        // 1 - диалог закрыт нажатием кнопки Cancel
        // 0- диалог закрыт нажатием кнопки OK = "event.Action.commit"
    }

    function DisplayValue(str)
    {
        var result = document.getElementById("result");
        result.innerHTML += str + "<br/>";
    }

    function onLoad()
    {
        DisplayValue(System.Gadget.name);
        DisplayValue(System.Gadget.version);

        System.Gadget.settingsUI = "Settings.html";
        System.Gadget.background = "Images/sky.jpg";
    }

</script>
</head>

<body onLoad="onLoad();">
    <span id="gadgetContent" style="font-family: Tahoma; font-size: 10pt;">
        <u>APIs Gadget Demo</u>
        <div id="result"></div>
    </span>
</body>
</html>

```

*Листинг: Settings.html*

```

<html>
<head>
    <style>
        body
        {

```

```

        width: 250px;
        height: 175px;
        font-family: Tahoma;
        font-size: 10px;
    }
</style>
<script>
    System.Gadget.onSettingsClosing = SettingsClosings;

    function SettingsClosing(event)
    {
        System.Debug.outputString(event.closeAction);
        // 1 - пользователь нажал кнопку Cancel = "event.Action.cancel"
        // 0 - пользователь нажал кнопку Cancel = "event.Action.commit!"
    }
</script>
</head>
<body>
    <label id="lbl">Settings page content</label><br/>
</body>
</html>

```

### System.Gadget.Settings

Метод	Описание
read	Читает параметр
readString	Читает параметр как строку
write	Записывает (сохраняет) параметры
writeString	Записывает параметры как строку

#### Пример

```

System.Gadget.Settings.write("age",30);
System.Gadget.Settings.write("PI_1", 3.14);
System.Gadget.Settings.writeString("PI_2","3.14");

```

```

System.Gadget.Settings.write("firstname", "John");
System.Gadget.Settings.writeString("lastname", "Smith");

```

```

var result = document.getElementById("result");
result.innerHTML = System.Gadget.Settings.readString("age") + "<br/>";
result.innerHTML += System.Gadget.Settings.read("PI_1") + "<br/>";

```

**Совет.** В общем случае для загрузки и сохранения числовых параметров следует использовать методы readString и writeString

### System.Shell

Метод Описание  
chooseFile  
chooseFolder  
exute  
saveFileDialog

Примеры:  
// выбор файла

```

var item = System.Shell.chooseFile(true,"Text File:*.txt:Png File:*.png" , ".", "");

// если пользователь выбрал файл
if ( item != null ) {
    result.innerHTML = "File selected: " + item.name + <br/>;
}

// выбор папки
var item = System.Shell.chooseFolder("Select a Folder", 0);
if ( item != null ) {
    result.InnerHTML += "Folder selected: " + item.name + "<br/>"

// запуск Notepad
System.Shell.execute ("notepad.exe");

```

## Электронная почта

Гаджет может получить доступ к электронной почте.

```

System.MessageStore
System.MessageStoreFolder
System.MessageStoreMessage

```

System.MessageStore объект возвращает коллекцию System.MessageStoreFolders объектов.

```

var result = document.getElementById("result");
var folders = System.MessageStore.Folders;
for ( i=0; i < folders.count; i++)
{
    var folder = folders.item(i);
    result.innerHTML += "Name: " + folder.name + "<br/>";
}

```

Результат:

```

Name: Inbox
Name: Outbox
Name: Sent Items
Name: Deleted Items
Name: Drafts
Name: Junk E-mail

```

Количество сообщений:

```

result.innerHTML += "Unread Messages: " + folder.unreadMessageCount;
result.innerHTML += "Messages: " + folder.messageCount

```

System.MessageStoreFolder объект возвращает коллекцию System.MessageStoreMessage объектов.

Свойства объекта System.MessageStoreMessage

Свойство	Значение
body	сообщение
from	от кого
subject	тема
to	кому

Пример

```

var result = document.getElementById("result");
var folders = System.MessageStore.Folders;
for ( i=0; i < folders.count; i++ )

```

```

{
    var folder = folders.items(i);
    result.innerHTML += "Name: " + folder.name + "<br/>";

    for ( j =0; j < folder.messageCount; j++ )
    {
        result.innerHTML += "Subject: " + folder.Messages.item(j).subject + "<br/>";
        result.innerHTML += "From: " + folder.Messages.item(j).from + "<br/>";
        result.innerHTML += "To: "" + folder.Messages.item(j).to + "<br/>";
        result.innerHTML += "Body: " + folder.Messages.item(j).body + "<br/>";
    }
}

```

## Контакты

Информация о контактах находится в папке C:\Users\UserName\Contacts. Доступ к информации, хранящейся в БД Контакты, обеспечивают объекты:

System.ContatactManager  
System.Contact

*Объект System.ContatactManager*

```

// получить доступ к контактам
var contacts = System.ContatacManager.Contacts;

// доступ к конкретному контакту
var contact = System.ContatacManager.Contacts.item(0);

// количество контактов
var count = System.ContatacManager.Contacts.count;

```

*Объект System.Contact*

Свойство	Значение
city	Город
country	
defaultEmail	
filePath	Путь к файлу vCard
homePhone	
mobilePhone	
name	
POBox	
postalCode	
state	
street	
workPhone	

Пример

```

var contacts = System.ContatacManager.Contacts;

var result = document.getElemetnById("result");

for (i=0; i < contacts.count; i++)
{
    result.innerHTML = <b> + contacts.item(i).name + "</b><br/>";
    result.innerHTML +=contacts.item(i).defaultEmail + "<br/>";
}

```

## System.Gadget.Flyout

Свойство	Описание
Document	
file	имя файла flyout-страницы
show	отображает flyout-страницу

Событие	Описание
onHide	
onShow	

### Пример

#### Листинг: APIs-Flyout.html

```
<html>
<head>
  <title>APIs Gadget Demo</title>
  <meta http-equiv ="Contetn-Type" content="text/html; charset=utf-8">
  <style>
    body
    {
      margin: 0;
      width: 130px;
      height: 260px;
    }
    #gadgetContent
    {
      width: 130px;
      height: 26px;
      top: 0px;
      text-align: center;
      font-family: Tahoma;
      font-size: 10 pt;
      position: absolute;
    }
  </style>
  <script>

    // --- определить процедуры обработки событий ---
    System.Gadget.Flyout.onHide = FlyoutHidden;
    System.Gadget.Flyout.onShow = FlyoutOnShow;

    // показать flyout-страницу
    function Flyout()
    {
      System.Gadget.Flyout.file = "flyout.html";
      System.Gadget.Flyout.show = true;
    }

    // обработка события onShow
    function FlyoutOnShow()
    {
      DisplayValue("Flyout shown");
    }

    // обработка события onHide
    function FlyoutHidden()
```

```

        {
            DisplayValue("Name entered: " +
                System.Gadget.Flyout.document.getElementById("txtName).value);
        }

// отображает строку в поле элемента result
function DisplayValue(str)
{
    var result = document.getElementById("result");
    result.innerHTML += str + "<br/>";
}

// обработка события Load
function onLoad()
{
    System.Gadget.bacground = "Images/sky.jpg";
}
</script>
<head>
<body onload="onLoad();">
    <span id="gadgetContent"; style="font-family: Tahoma; font-size: 10pt;">
        <u>APIs Gadget Demo</u><br/>
        <div id="result"></div>
        <input type="button" value="Flyout" onclick="Flyout();" />
    </span>
</body>
</html>

```

Листинг: flyout.html

```

<html>
<head>
    <style>
        body
        {
            width: 200px;
            height: 100px;
            font-family: Tahoma;
            font-size: 12pt;
        }
        table
        {
            font-family: Tahoma;
            font-size: 12pt;
        }
    </style>
    <script>
        // страница загружена
        function done()
        {
            System.Gadget.Flyout.show = false;
        }
    </script>
</head>
<body>
    <label>Flyout page</label><br />
    Enter your name:
    <input type="text" name="textName" id="txtName" />
    <input type="button" value="Done" onclick="done();" />
</body>

```

</html>

**Звук**

```
System.Sound.beep();
```

```
System.Sound.playSound(@"\windows\media\windows Critical Stop.wav");
```