

Scripted Inventory System

GarageGames.com

Author: [befly](#) (c++programmer)**NEW!** Download / view this tutorial as [PDF!](#)

This inventory gui system is a quite simple extension to the script-only inventory system in Torque.

First of all, here is a pic of the InventoryManager to show what I am talking about here:



The concept/idea is taken from **Tim "Spock" Newell's InventoryManager**, but mine is totally scripted and an extension to the standard inventory scripts already available in Torque - you will need to touch the engine for some advanced features, though, e.g. for the 3D object view (for an explanation how to integrate that, see [this tutorial](#)). You will also need

BitmapButton controls (I've used the resource by Tim "Spock" Newell for that). Other than that, it's all in the scripts. And while Spock's InvManager is single-player only, mine works in multiplayer, too... :)

The gui has 11 inventory slots, if your player carries more than that, you can scroll up and down with the arrow buttons. It shows the items and the amounts on the left, a 3D view of the item in the upper right and the item description in the lower right.

I'm going to add some more explanations here if I find some time, for now I've only pasted the relevant changes/additions here... if you've got any specific question, feel free to send me an email or post it in the forums... :)

The very basic concept is that I simply store an array with item names and the total number of inventory items for each individual player (%client.player.totalInvName[<id>], %client.player.totalInvNum, %client.player.totalInvDirectUseName, %client.player.totalInvDirectUseNum). The "...DirectUseName" array stores all the items that should be accessible via a shortcut key (see below).

Ok, so here we go:

Add these functions somewhere in "**commands.cs**":

```
// file: server/scripts/commands.cs
////////////////////////////////////
// Inventory functions
////////////////////////////////////
function serverCmdShiftInventory(%client,%direction)
{
    %player = %client.player;
    %totalInvCount = %player.totalInvNum;
    if(%direction $= "up")
        commandToClient(%client, 'ScrollInventoryUp',%player,%totalInvCount);
    else
        commandToClient(%client, 'ScrollInventoryDown',%player,%totalInvCount);
}

function serverCmdDoShift(%client,%bottomSlot)
{
    %myplayer = %client.player;
    %counter = 10;
    %directUseCounter = 0;
    for(%i = 1; %i <= 11; %i++)
    {
        %usable = 0;
        %key = 0;
        %invName = %myplayer.totalInvName[%bottomSlot-%counter];
        %amountText = %myplayer.getInventory(%invName);
        %totalInvCount = %myplayer.totalInvDirectUseNum;
        for(%j = 1; %j <= %totalInvCount; %j++)
        {
            %directUseName = %myplayer.totalInvDirectUseName[%j];
            if(%invName $= %directUseName)
            {
                %key = %j;
            }
        }
    }
}
```

```
                %usable = 1;
            }
        }
        commandToClient(%client, 'UpdateInventoryScrolling',
            %amountText, %key, %usable, %i, %bottomSlot-%counter, %invName);
        %counter--;
    }
}

function serverCmdDisplayInventory(%client)
{
    %player = %client.player;
    //Send the info to the client to display it
    commandToClient(%client, 'PopInventory', %player);
}

function serverCmdListInventory(%client, %maxNum)
{
    %player = %client.player;
    %totalInvCount = %player.totalInvNum;
    %directUseCounter = 0;
    for(%i = 1; %i <= %totalInvCount; %i++)
    {
        if(%i > %maxNum)
            return;
        %usable = 0;
        %invName = %player.totalInvName[%i];
        %invDirectUse = %invName.directUse;
        %invNumber = %player.getInventory(%invName);
        %currSlot = "Slot" @ %i;
        %currAmountSlot = "Amount" @ %i;
        if(%invDirectUse)
        {
            %directUseCounter++;
            %usable = 1;
        }
        commandToClient(%client, 'UpdateInventoryGui', %currSlot, %currAmountSlot,
            %i, %invName, %invNumber, %directUseCounter, %usable);
    }
}

function serverCmdClearInventory(%client, %maxNum)
{
    error("serverCmdClearInventory() called for" SPC %client);
    %player = %client.player;
    %totalInvCount = %player.totalInvNum;
    for(%i = 1; %i <= %totalInvCount; %i++)
    {
```

```

        if(%i > %maxNum)
            return;
        %currSlot = "Slot" @ %i;
        %currAmountSlot = "Amount" @ %i;
        commandToClient(%client, 'ClearInventoryGui', %currSlot, %currAmountSlot);
    }
}

function serverCmdViewItem(%client, %num)
{
    %invName = %client.player.totalInvName[%num];
    %shapeFile = %invName.shapeFile;
    %displayName = %invName.pickUpName;
    commandToClient(%client, 'UpdatePreview', %shapeFile, %displayName, %invName);
}

```

The next part requires changes/additions to **"inventory.cs"** - simply add/replace the following functions:

```

// file: server/scripts/inventory.cs
//-----
// Inventory server commands
//-----

function serverCmdUse(%client,%data)
{
    %control = %client.getControlObject();
    %control.use(%data);
    if (%data.category $= "Weapon" && %client.player.getInventory(%data) > 0)
    {
        commandToClient(%client, 'UpdateQuickInv',%data.shapeFile);
    }
}

function serverCmdUseById(%client,%id)
{
    %control = %client.getControlObject();
    if(%id > %client.player.totalInvDirectUseNum)
        return;
    %control.useById(%id);
    %itemName = %client.player.totalInvDirectUseName[%id];
    if (%client.player.getInventory(%itemName) > 0)
    {
        commandToClient(%client,'UpdateQuickInv',%itemName.shapeFile);
    }
}

```

```
//-----  
// ShapeBase inventory support  
//-----  
  
//-----  
  
function ShapeBase::use(%this,%data)  
{  
    // Use an object in the inventory.  
    if (%this.getInventory(%data) > 0)  
    {  
        return %data.onUse(%this);  
    }  
    return false;  
}  
function ShapeBase::useById(%this,%id)  
{  
    // Use an object in the inventory.  
    %itemName = %this.totalInvDirectUseName[%id];  
    if (%this.getInventory(%itemName) > 0)  
    {  
        return %itemName.onUse(%this);  
    }  
    return false;  
}  
  
...  
  
function ShapeBase::incInventory(%this,%data,%amount)  
{  
    // Increment the inventory by the given amount. The return value  
    // is the amount actually added, which may be less than the  
    // requested amount due to inventory restrictions.  
    %max = %this.maxInventory(%data);  
    %total = %this.inv[%data.getName()];  
  
    %totalInvCount = %this.totalInvNum;  
    %itemAlreadyInInventory = false;  
    for(%i = 1; %i <= %totalInvCount; %i++)  
    {  
        %invName = %this.totalInvName[%i];  
        %invNumber = %this.getInventory(%invName);  
        if(%data.getName() $= %invName){  
            %itemAlreadyInInventory = true;  
        }  
    }  
    if(!%itemAlreadyInInventory)
```

```

    {
        %this.totalInvName[%this.totalInvNum++] = %data.getName();
        if(%data.directUse $= "1")
        {
            %this.totalInvDirectUseName[%this.totalInvDirectUseNum++] = %data.getName();
        }
    }

    if (%total < %max) {
        if (%total + %amount > %max)
            %amount = %max - %total;
        %this.setInventory(%data,%total + %amount);
        return %amount;
    }
    return 0;
}

```

I've added commands to use the items by their ID in "**client/config.cs**":

```

moveMap.bindCmd(keyboard, "1", "commandToServer('\useById\',"1\");", "");
moveMap.bindCmd(keyboard, "2", "commandToServer('\useById\',"2\");", "");
moveMap.bindCmd(keyboard, "3", "commandToServer('\useById\',"3\");", "");
moveMap.bindCmd(keyboard, "4", "commandToServer('\useById\',"4\");", "");
moveMap.bindCmd(keyboard, "5", "commandToServer('\useById\',"5\");", "");
moveMap.bindCmd(keyboard, "6", "commandToServer('\useById\',"6\");", "");
moveMap.bindCmd(keyboard, "7", "commandToServer('\useById\',"7\");", "");
moveMap.bindCmd(keyboard, "8", "commandToServer('\useById\',"8\");", "");
moveMap.bindCmd(keyboard, "9", "commandToServer('\useById\',"9\");", "");

```

so that if you press "1" the player uses whatever is in slot 1 of the gui...

Important:

To support this "shortcut" system, you have to add one line to each item/weapon/etc. you want to have accessible via the shortcut keys (1-9) - add this line into the appropriate "**ItemData**" datablock:

```

directUse = 1;

```

so for example:

```

datablock ItemData(Rifle)
{
    // Mission editor category
    category = "Weapon";

    // beffy: hook for InventoryManager
    directUse = 1;
}

```

To be able to open the gui, add this bind to "**client/scripts/default.bind.cs**" and "**client/init.cs**":

```
moveMap.bind(keyboard, "i", toggleInventory );
```

and this function to "**client/scripts/default.bind.cs**":

```
function toggleInventory(%val) {  
    if (%val) {  
        commandToServer('DisplayInventory');  
    }  
}
```

After that, open **gui/guiBitmapCtrl.h** and add this on top:

```
#ifndef _GUIMOUSEEVENTCTRL_H_  
#include "gui/guiMouseEventCtrl.h"  
#endif
```

and also change

```
class GuiBitmapCtrl : public GuiCtrl  
{  
private:  
    typedef GuiCtrl Parent;
```

to

```
class GuiBitmapCtrl : public GuiMouseEventCtrl  
{  
private:  
    typedef GuiMouseEventCtrl Parent;
```

so that your gui bitmaps trigger **mouse events** and you can use bitmap arrows for scrolling...

Now, do the same for "**gui/guiTextCtrl.h**" to get mouse events for the text entries, too!

This is the main file for the InventoryManager. Add it to your "**client/scripts**" folder and exec it in "**client/init.cs**":

Please note:

This gui has eleven slots "hardcoded", so if you want more or less slots, some parts have to be adjusted...

```
// file: client/scripts/InventoryGui.cs
//-----
// InventoryGui.cs
//-----

$CurrentBottomSlot = 11;

new ActionMap(InvMap);
InvMap.bindCmd(keyboard, "i", "closeInventory();", "");

function closeInventory()
{
    Canvas.setContent(PlayGui);
}

function InventoryGui::onWake()
{
    displayWindow.setEmpty();
    DisplayText.setText("");

    $enableDirectInput = "1";
    activateDirectInput();

    InvMap.push();
}

function InventoryGui::onSleep()
{
    InvMap.pop();
    commandToServer('ClearInventory');
}

function InventoryGui::onRemove()
{
    error("InventoryGui::onRemove() called!");
    // clear the text fields
    commandToServer('ClearInventory');
}

function clientCmdPopInventory(%player)
{
    Canvas.setContent( InventoryGui );
    commandToServer('listInventory',"11");
}

function clientCmdUpdateInventoryGui(%currSlot,%currAmountSlot,%i,%invName,%invNumber)
{
    %currSlot.setText(%i SPC "-" SPC %invName);
}
```

```
        %currAmountSlot.setText(%invNumber);
    }
function clientCmdClearInventoryGui(%currSlot,%currAmountSlot)
{
    %currSlot.setText("");
    %currAmountSlot.setText("");
}

function clientCmdUpdatePreview(%shapeFile, %displayName, %invName)
{
    if(%shapeFile != "")
    {
        displayWindow.setModel(%shapeFile, "");
    }
    if(%displayName != "")
    {
        DisplayText.setText(%displayName);
    }
    commandToServer('use',%invName);
}
//-----
// Display function
//-----
function DisplayItem(%num)
{
    commandToServer('ViewItem',%num);
}
function clientCmdUpdateQuickInv(%shape)
{
    QuickInvSlot.setModel(%shape, "");
}

//-----
// Slot click events
//-----
// Top Slots
function Slot1::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 10);
}
function Slot2::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 9);
}
function Slot3::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 8);
}
function Slot4::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 7);
}
```

```
}
function Slot5::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 6);
}
function Slot6::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 5);
}
function Slot7::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 4);
}
function Slot8::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 3);
}
function Slot9::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 2);
}
function Slot10::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot - 1);
}
function Slot11::onMouseDown(%this, %obj) {
    DisplayItem($CurrentBottomSlot);
}

//-----
// "Arrow Keys"
//-----

function UpArrow::onMouseDown(%this, %obj)
{
    commandToServer('ShiftInventory', "up");
}
function DownArrow::onMouseDown(%this, %obj)
{
    commandToServer('ShiftInventory', "down");
}

//-----
// Shift - shifts the slots
//-----
function clientCmdScrollInventoryUp(%myplayer, %invCount)
{
    if (%invCount > 10 && $CurrentBottomSlot > 11) {
        $CurrentBottomSlot--;
        commandToServer('DoShift', $CurrentBottomSlot);
    }
}
}
```

```
function clientCmdScrollInventoryDown(%myplayer, %invCount)
{
    if (%invCount > 10 && $CurrentBottomSlot < (%invCount)) {
        $CurrentBottomSlot++;
        commandToServer('DoShift', $CurrentBottomSlot);
    }
}

function clientCmdUpdateInventoryScrolling(%amount, %key, %usable, %num, %invNum, %invName)
{
    %amountSlot = "Amount" @ %num;
    %invSlot = "Slot" @ %num;
    if(%usable)
    {
        %amount = %amount SPC "    Key:" SPC %key;
    }
    %invSlot.setText(%invNum SPC %invName);
    %amountSlot.setText(%amount);
}
```

clientCmdUpdateQuickInv() shows a little 3D view of the current selected item in the PlayGui, to get this, simply add this snipit somewhere in your PlayGUI:

```
new GuiObjectView(QuickInvSlot) {
    profile = "GuiDefaultProfile";
    horizSizing = "right";
    vertSizing = "top";
    position = "180 430";
    extent = "62 56";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    cameraZRot = "0";
    forceFOV = "0";
};
```

Here's how it looks like:



Finally, here is the complete .gui file. Note that I've used the BitmapButtonControls by Tim "Spock" Newell for the Up- and Down arrows. So if you want the same scrolling functionality, you're going to have to put that in, too (involves engine changes). Add that file to "**client/ui**" for example and exec it in "**client/init.cs**", too. You will need a background image and two arrow images, of course (replace the bitmap names used here appropriately).

```
//--- OBJECT WRITE BEGIN ---
new GameTSCtrl(InventoryGui) {
  profile = "InventoryGuiProfile";
  horizSizing = "right";
  vertSizing = "bottom";
  position = "0 0";
  extent = "640 480";
  minExtent = "8 8";
  visible = "1";
  helpTag = "0";
  cameraZRot = "0";
  forceFOV = "0";

  new GuiBitmapCtrl(InventoryPanel) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "80 52";
    extent = "508 352";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    bitmap = "./InvPanel";
    wrap = "0";
  };
  new GuiBitmapButtonCtrl(UpArrow) {
    profile = "BitmapButtonProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "307 59";
    extent = "36 32";
    minExtent = "8 8";
    visible = "1";
```

```
    helpTag = "0";
    lockMouse = "0";
    bitmap = "./UpArrow";
    HighlightBitmap = "./UpArrow";
    wrap = "0";
};
new GuiBitmapButtonCtrl(DownArrow) {
    profile = "BitmapButtonProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "308 359";
    extent = "36 32";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    bitmap = "./DownArrow";
    HighlightBitmap = "./DownArrow";
    wrap = "0";
};
new GuiTextCtrl(InvpName) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "280 12";
    extent = "148 36";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiObjectView(DisplayWindow) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "380 64";
    extent = "200 200";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    cameraZRot = "0";
    forceFOV = "0";
};
new GuiTextCtrl(DisplayText) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
```

```
        vertSizing = "relative";
        position = "400 256";
        extent = "160 160";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Slot1) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "108 64";
        extent = "100 24";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Slot2) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "108 88";
        extent = "100 24";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Slot3) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "108 112";
        extent = "100 28";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Slot4) {
        profile = "InventoryGuiProfile";
```

```
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 140";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot5) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 164";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot6) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 188";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot7) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 212";
    extent = "100 28";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot8) {
```

```
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 240";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot9) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 264";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot10) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 288";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Slot11) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "108 312";
    extent = "100 28";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
```

```
new GuiTextCtrl(Amount1) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 64";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount2) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 88";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount3) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 112";
    extent = "100 28";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount4) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 140";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
```

```
};
new GuiTextCtrl(Amount5) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 164";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount6) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 188";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount7) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 212";
    extent = "100 28";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
    maxLength = "255";
};
new GuiTextCtrl(Amount8) {
    profile = "InventoryGuiProfile";
    horizSizing = "relative";
    vertSizing = "relative";
    position = "224 240";
    extent = "100 24";
    minExtent = "8 8";
    visible = "1";
    helpTag = "0";
    lockMouse = "0";
```

```
        maxLength = "255";
    };
    new GuiTextCtrl(Amount9) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "224 264";
        extent = "100 24";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Amount10) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "224 288";
        extent = "100 24";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
    new GuiTextCtrl(Amount11) {
        profile = "InventoryGuiProfile";
        horizSizing = "relative";
        vertSizing = "relative";
        position = "224 312";
        extent = "100 28";
        minExtent = "8 8";
        visible = "1";
        helpTag = "0";
        lockMouse = "0";
        maxLength = "255";
    };
};
//--- OBJECT WRITE END ---
```

[<< Previous](#)[Table of Contents](#)[Author:beffy](#)[Next >>](#)