

# Asimplespell

GarageGames.com



DocVersion1.0.x

Author: beffy(c++-programmer)

Okay, this is a very simple way to "cast spells", maybe you can improve/extend it to get something like a "magic system" or at least a little "spell gui" ...

First of all, thank to Sabrecy and LabRat and "the IRC guys" for helping me to solve some problems!

Ok, let's start with the actual "doSpell()" function, which is placed in "fps/server/scripts/game.cs" (shouldn't really matter where you put it on the server side...). It takes the radius, the damage, should be done in, the client casting the spell, and the type of spell (currently "1" for Explosions, "2" for ParticleEmitters only... here it is:

```
// fps/server/scripts/game.cs
// NOTE: if you encounter problems in Multiplayer, try to put this on the client side as
// clientCmdDoSpell - didn't test it yet in MP...

function doSpell(%radius,%client,%type)
{
    %radiusDamage = 8.0;
    %pos = %client.player.getTransform();
    %x = getWord(%pos, 0);
    %y = getWord(%pos, 1);
    %z = getWord(%pos, 2);
    // adjust z value a bit...
    %z += 2.0;
    %finalPos = %x SPC %y SPC %z;

    %eye = %client.player.getEyeVector();
    %vec = vectorScale(%eye, 20);
    %finalPos = vectorAdd(%finalPos, %vec);

    switch(%type)
    {
        case "1":
            // EXPLOSION
            %p = new explosion() {
                dataBlock = "SpellExplosion";
                position = %finalPos;
            };
            MissionCleanup.add(%p);
            // Radius damage is a support script defined in radiusDamage.cs
            radiusDamage(%p,%finalPos,%radius,%radiusDamage,"SpellBolt",0);
    }
}
```

```

case "2":
// PARTICLES WITH SOUND
%p = new ParticleEmitterNode() {
    position = %finalPos;
    rotation = "1 0 0 0";
    scale = "1 1 1";
    dataBlock = "defaultParticleEmitterNode";
    emitter = "BlueSpellEmitter";
    velocity = "1";
};
serverPlay3D(SpellSound,%client.player);
radiusDamage(%p,%finalPos,%radius,%radiusDamage,"SpellBolt",0);
%p.schedule(2000,"delete");
default:
// EXPLOSION
%p = new explosion() {
    dataBlock = "SpellExplosion";
    position = %finalPos;
};
MissionCleanup.add(%p);
radiusDamage(%p,%finalPos,%radius,%radiusDamage,"SpellBolt",0);
}
}

```

It takes the players' transform (position), adjusts the value to get it off the ground, then takes the current eye vector of the player, scales it (adjust this value to change the distance of the spell) and adds it to the transform vector - otherwise the spell would just stay in the same position all the time, even if the player was turning... If it only uses the ParticleEmitter, it also plays a sound using:

```
serverPlay3D(SpellSound,%client.player);
```

Then \!vegocalltothefunctionin"fps/server/scripts/commands.cs":

```
// fps/server/scripts/commands.cs
function serverCmdDoSpell(%client)
{
    doSpell("50",%client,1);
}

```

The actual call to the function is in "fps/client/scripts/defaultBind.cs":

```
function castSpell(%val)
{
    if(%val)
    {
        commandToServer('DoSpell');
    }
}

```

```

    }
}

// bind a key to it
moveMap.bind(keyboard, "ctrl s", castSpell);

You also need to add this mapping in "fps/client/config.cs":

moveMap.bind(keyboard, "ctrl s", castSpell);

```

If you also want this to be in the options dialog so that you can remap it, put this into the remapping section in "fps/client/scripts/optionsDlg.cs":

```

// fps/client/scripts/optionsDlg.cs
$RemapName[$RemapCount] = "Cast Spell";
$RemapCmd[$RemapCount] = "castSpell";
$RemapCount++;

```

Okay, then I've made a new explosion file by simply copying the important stuff from "crossbow.cs" (RealmWars), this file introduces some AudioProfiles, ParticleEmitter, Explosions, and a, like "virtual projectile", the "SpellBolt"; -, which simply need to be applied to damage by using the spell... here is just the AudioProfile for the spell itself, which I also use for the "ParticleEmitter" spell type... for further details, please look into the file which is contained in the zip file provided...

```

// fps/server/scripts/spellExplosion.cs

datablock AudioProfile(SpellSound)
{
    fileName = "~/data/sound/fx/spark1.wav";
    description = AudioClose3d;
    preload = true;
};
...
// all the other explosion/emitter stuff...

```

Ok, here is the ParticleEmitter which I use as type "2" spell... It's just an alternative if you don't want a complete explosion to appear... I've put it into my own "customParticles" file (which is executed in "game.cs", of course...)

```

// fps/server/scripts/customParticles.cs
datablock ParticleData(BlueSpellParticle)
{
    dragCoefficient = 1.11437;
    gravityCoefficient = -0.735043;
    windCoefficient = 0;
    inheritedVelFactor = 0.483366;
    constantAcceleration = 0;
    lifetimeMS = 1056;
}

```

```

lifetimeVarianceMS = 256;
useInvAlpha = 0;
spinRandomMin = -159;
spinRandomMax = 172;
textureName = "fps/data/shapes/rifle/smokeParticle";
times[0] = 0;
times[1] = 1;
colors[0] = "0.002362 0.010866 0.700000 0.370079";
colors[1] = "0.000000 0.102362 0.800000 0.740157";
sizes[0] = 6.08863;
sizes[1] = 0;
};

dataBlock ParticleEmitterData(BlueSpellEmitter)
{
    ejectionPeriodMS = 10;
    periodVarianceMS = 2;
    ejectionVelocity = 2.75;
    velocityVariance = 1.62;
    ejectionOffset = 0;
    thetaMin = 47;
    thetaMax = 90;
    phiReferenceVel = 144;
    phiVariance = 360;
    overrideAdvances = 0;
    orientParticles = 0;
    orientOnVelocity = 1;
    particles = "BlueSpellParticle";
};

```

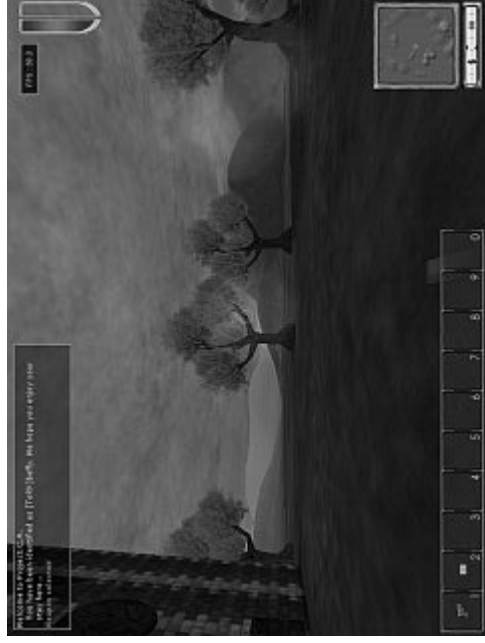
This could be improved in various directions, of course...

- First of all, I don't really like the switch block choosing between the 2 (or more) spell types... maybe you can come up with some better random dynamics solution here...
- Then, the distance is always the same, so the spell isn't "spreading"...
- Also, there is no collision detection involved... so the spell can appear inside of trees, buildings, in the terrain, etc... should be added, too...;-)

Anyhow, I hope it's something you can use to play around with and may be improve it and give it back to the community...;-)

You can download the sandwav files needed here!

As always, here are some screenshots:



<<Previous

TableofContents

Author:beffy

Next>>