

xp_aspectManager

functions Guide

v1.0 for v1.4 – 2024-05-12 by Septirage

This document will try to compile all functions added by xp_aspectManager, trying to give you useful informations and help on "how to" use them. It will be updated regularly.

- [xp_aspectManager.nss](#)
 - [xp_aspectManager_creature.nss](#)
 - [xp_aspectManager_area.nss](#)
 - [xp_aspectManager_item.nss](#)
 - [xp_aspectManager_light.nss](#)
-
-

xp_aspectManager.nss

- [Hit Points Functions](#)
 - [SetCurrentHitPoints xpAM](#)
 - [GetCurrentHitPoints xpAM](#)
 - [SetTemporaryHitPoints xpAM](#)
 - [GetTemporaryHitPoints xpAM](#)
 - [SetMaxHitPoints xpAM](#)
 - [GetMaxHitPoints xpAM](#)
 - [SetModMaxHitPoints xpAM](#)
 - [GetModMaxHitPoints xpAM](#)
- [Appearance](#)
 - [SetAppearance xpAM](#)
 - [GetAppearance xpAM](#)
- [UVScroll](#)
 - [SetUVScroll xpAM](#)
 - [GetUVScroll xpAM](#)
 - [SetUScroll xpAM](#)
 - [GetUScroll xpAM](#)
 - [SetVScroll xpAM](#)
 - [GetVScroll xpAM](#)
- [Inventory](#)
 - [SetHasInventory xpAM](#)
 - [GetHasInventory xpAM](#)
 - [SetInventoryMaxSize xpAM](#)
 - [GetInventoryMaxSize xpAM](#)
 - [GetInventoryNbItem xpAM](#)
- [Lock](#)
 - [SetAutoRemoveKey xpAM](#)
 - [GetAutoRemoveKey xpAM](#)
- [Trigger](#)
 - [GetTriggerType xpAM](#)
 - [SetTriggerType xpAM](#)
 - [SetTriggerGeometry xpAM](#)
- [Transition](#)
 - [SetIsInvisibleTransition xpAM](#)
 - [GetIsInvisibleTransition xpAM](#)
 - [SetIsPartyTransition xpAM](#)
 - [GetIsPartyTransition xpAM](#)
 - [SetTransitionLinkObjectType xpAM](#)
 - [GetTransitionLinkObjectType xpAM](#)
 - [SetLoadScreenID xpAM](#)
 - [GetLoadScreenID xpAM](#)
 - [SetTransitionDestination xpAM](#)
 - [GetTransitionDestination xpAM](#)
- [Saves](#)
 - [SetMagicResistance xpAM](#)
 - [GetMagicResistance xpAM](#)
 - [SetWillSavingThrow xpAM](#)

- [SetFortitudeSavingThrow_xpAM](#)
- [SetReflexSavingThrow_xpAM](#)
- [Faction](#)
 - [GetFactionID_xpAM](#)
 - [GetIsBaseFaction_xpAM](#)
- [Visual](#)
 - [SetAppearanceSEF_xpAM](#)
 - [GetAppearanceSEF_xpAM](#)
 - [SetColor_xpAM](#)
 - [GetColor_xpAM](#)
- [Position and facing](#)
 - [SetFacing_xpAM](#)
 - [SetPosition_xpAM](#)
 - [SetPositionAndFacing_xpAM](#)
- [Music](#)
 - [MusicBackgroundDayForPC_xpAM](#)
 - [MusicBackgroundNightForPC_xpAM](#)
 - [MusicBattleForPC_xpAM](#)
 - [MusicBackgroundRunForPC_xpAM](#)
 - [MusicBattleRunForPC_xpAM](#)
- [Refresh, View & Collision](#)
 - [RefreshCreatureForPC_xpAM](#)
 - [RefreshCreature_xpAM](#)
 - [RefreshEquipedItemForPC_xpAM](#)
 - [RefreshEquipedItem_xpAM](#)
 - [RefreshObjectForPC_xpAM](#)
 - [RefreshObject_xpAM](#)
 - [ShowObject_xpAM](#)
 - [ShowObjectForPC_xpAM](#)
 - [MaskObject_xpAM](#)
 - [MaskObjectForPC_xpAM](#)
 - [SetObjectDynamicCollision_xpAM](#)
 - [GetObjectDynamicCollision_xpAM](#)
- [Constants](#)
 - [Light Color Types \(LIGHT COLOR TYPE *\)](#)
 - [Transition Types \(TRANSITION TYPE *\)](#)
 - [Flicker Types \(FLICKER TYPE *\)](#)
 - [Armor Model Parts \(ARMOR MODEL PART *\)](#)
 - [Trigger Types \(TRIGGER TYPE *\)](#)

Hit Points Functions

SetCurrentHitPoints_xpAM

```
void SetCurrentHitPoints_xpAM(object oObject, int iHP);
```

Set CurrentHitPoints of oObject to iHP.

Parameters

- `oObject` - The object to set the CurrentHitPoints for.
- `iHP` - The CurrentHitPoints to set.

Note

This is the Current "base" HitPoints, without the possible Temporary HitPoints

GetCurrentHitPoints_xpAM

```
int GetCurrentHitPoints_xpAM(object oObject);
```

Get CurrentHitPoints of oObject.

Parameters

- `oObject` - The object to get the CurrentHitPoints for.

Return Value

The CurrentHitPoints of the object.

Note

This is the Current "base" HitPoints, without the possible Temporary HitPoints

SetTemporaryHitPoints_xpAM

```
void SetTemporaryHitPoints_xpAM(object oObject, int iHP);
```

Set Temporary HitPoints of oObject to iHP.

Parameters

- `oObject` - The object to set the Temporary HitPoints for.
 - `iHP` - The Temporary HitPoints to set.
-

GetTemporaryHitPoints_xpAM

```
int GetTemporaryHitPoints_xpAM(object oObject);
```

Get Temporary HitPoints of oObject.

Parameters

- `oObject` - The object to get the Temporary HitPoints for.

Return Value

The Temporary HitPoints of the object.

SetMaxHitPoints_xpAM

```
void SetMaxHitPoints_xpAM(object oObject, int iHP);
```

Set Max HitPoints of oObject to iHP.

Parameters

- `oObject` - The object to set the Max HitPoints for.
 - `iHP` - The Max HitPoints to set.
-

GetMaxHitPoints_xpAM

```
int GetMaxHitPoints_xpAM(object oObject);
```

Get Max HitPoints of oObject.

Parameters

- `oObject` - The object to get the Max HitPoints for.

Return Value

The Max HitPoints of the object.

SetModMaxHitPoints_xpAM

```
void SetModMaxHitPoints_xpAM(object oObject, int iHP);
```

Set the Modifier value of Max HitPoints of oObject to iHP.

Parameters

- `oObject` - The object to set the Modifier value of Max HitPoints for.
 - `iHP` - The Modifier value to set.
-

GetModMaxHitPoints_xpAM

```
int GetModMaxHitPoints_xpAM(object oObject);
```

Get the Modifier value of Max HitPoints of oObject.

Parameters

- `oObject` - The object to get the Modifier value of Max HitPoints for.

Return Value

The Modifier value of Max HitPoints of the object.

Appearance

SetAppearance_xpAM

```
void SetAppearance_xpAM(object oObject, int iApp);
```

Set Appearance of oObject to iApp. Will be based on respective 2DA tables. Work for Door or Placeable.

Parameters

- `oObject` - The object to set the Appearance for. (Door or Placeable)
 - `iApp` - The corresponding 2DA index of the appearance to set.
-

GetAppearance_xpAM

```
int GetAppearance_xpAM(object oObject);
```

Get Appearance of oObject. Will be related to respective 2DA tables. Work for Door or Placeable.

Parameters

- `oObject` - The object to get the Appearance for. (Door or Placeable)

Return Value

The corresponding 2DA index of the appearance of the object.

UVScroll

SetUVScroll_xpAM

```
void SetUVScroll_xpAM(object oObject, int bActive);
```

Enable or Disable UVScroll for oObject. Work for placeable.

Parameters

- `oObject` - The object to enable or disable UVScroll for. (Placeable only)
 - `bActive` - The flag to enable (TRUE) or disable (FALSE) UVScroll.
-

GetUVScroll_xpAM

```
int GetUVScroll_xpAM(object oObject);
```

Get the activation status of UVScroll for oObject. Work for placeable.

Parameters

- `oObject` - The object to get the UVScroll activation status for. (Placeable only)

Return Value

The activation status of UVScroll for the object (TRUE if enabled, FALSE if disabled).

SetUScroll_xpAM

```
void SetUScroll_xpAM(object oObject, float fUScroll);
```

Set UScroll speed for oObject to fUScroll. Work for placeable.

Parameters

- `oObject` - The object to set the UScroll speed for. (Placeable only)
 - `fUScroll` - The UScroll speed to set.
-

GetUScroll_xpAM

```
float GetUScroll_xpAM(object oObject);
```

Get UScroll speed for oObject. Work for placeable.

Parameters

- `oObject` - The object to get the UScroll speed for. (Placeable only)

Return Value

The UScroll speed of the object.

SetVScroll_xpAM

```
void SetVScroll_xpAM(object oObject, float fVScroll);
```

Set VScroll speed for oObject to fVScroll. Work for placeable.

Parameters

- `oObject` - The object to set the VScroll speed for. (Placeable only)
 - `fVScroll` - The VScroll speed to set.
-

GetVScroll_xpAM

```
float GetVScroll_xpAM(object oObject);
```

Get VScroll speed for oObject. Work for placeable.

Parameters

- `oObject` - The object to get the VScroll speed for. (Placeable only)

Return Value

The VScroll speed of the object.

Inventory

SetHasInventory_xpAM

```
void SetHasInventory_xpAM(object oObject, int bHasInventory);
```

Enable or Disable the inventory for oObject Placeable.

Parameters

- `oObject` - The Placeable to enable or disable the inventory for.
 - `bHasInventory` - The flag to enable (TRUE) or disable (FALSE) the inventory.
-

GetHasInventory_xpAM

```
int GetHasInventory_xpAM(object oObject);
```

Return the value of HasInventory Flag for oObject Placeable.

Parameters

- `oObject` - The Placeable to get the HasInventory flag for.

Return Value

The value of the HasInventory flag for the object (TRUE if enabled, FALSE if disabled).

SetInventoryMaxSize_xpAM

```
void SetInventoryMaxSize_xpAM(object oObject, int iMaxSize);
```

Set the MaximumInventorySize of Inventory for oObject Placeable to iMaxSize.

Parameters

- `oObject` - The object to set the MaximumInventorySize for.
- `iMaxSize` - The MaximumInventorySize to set.

Note

iMaxSize max value is 142.

GetInventoryMaxSize_xpAM

```
int GetInventoryMaxSize_xpAM(object oObject);
```

Get the MaximumInventorySize of Inventory for oObject Placeable.

Parameters

- `oObject` - The object to get the MaximumInventorySize for.

Return Value

The MaximumInventorySize of the object.

GetInventoryNbItem_xpAM

```
int GetInventoryNbItem_xpAM(object oObject);
```

Get the number of objects in the oObject inventory. Work for placeable.

Parameters

- `oObject` - The object to get the number of inventory items for.

Return Value

The number of items in the inventory of the object.

Lock

SetAutoRemoveKey_xpAM

```
void SetAutoRemoveKey_xpAM(object oObject, int bAutoRemove);
```

Set the AutoRemoveKey Flag value for oObject. Work for Placeable, Door.

Parameters

- `oObject` - The object to set the AutoRemoveKey flag for.
 - `bAutoRemove` - The flag to enable (TRUE) or disable (FALSE) AutoRemoveKey.
-

GetAutoRemoveKey_xpAM

```
int GetAutoRemoveKey_xpAM(object oObject);
```

Get the AutoRemoveKey Flag value for oObject. Work for Placeable, Door.

Parameters

- `oObject` - The object to get the AutoRemoveKey flag for.

Return Value

The value of the AutoRemoveKey flag for the object (TRUE if enabled, FALSE if disabled).

Trigger

GetTriggerType_xpAM

```
int GetTriggerType_xpAM(object oObject);
```

Get the trigger type (return a [TRIGGER_TYPE *](#) constant).

Parameters

- `oObject` - The object to get the trigger type for.

Return Value

The trigger type as a [TRIGGER_TYPE *](#) constant.

SetTriggerType_xpAM

```
void SetTriggerType_xpAM(object oObject, int iType);
```

Set trigger type, use [TRIGGER_TYPE *](#) constant.

Parameters

- `oObject` - The object to set the trigger type for.
 - `iType` - The trigger type as a [TRIGGER_TYPE *](#) constant.
-

SetTriggerGeometry_xpAM

```
void SetTriggerGeometry_xpAM(object oTrigger, string sNewGeometry);
```

Set a new Geometry for oTrigger.

Parameters

- `oTrigger` - The trigger object to set the geometry for.
 - `sNewGeometry` - The new geometry in the format "NumOfPoint Point1X Point1Y Point1Z Point2X Point2Y Point2Z ...".
-

Transition

SetIsInvisibleTransition_xpAM

```
void SetIsInvisibleTransition_xpAM(object oObject, int bInvisibleTransition);
```

Set the Invisible Transition flag for oObject. Work for Door.

Parameters

- `oObject` - The object to set the Invisible Transition flag for.
 - `bInvisibleTransition` - The flag to enable (TRUE) or disable (FALSE) Invisible Transition.
-

GetIsInvisibleTransition_xpAM

```
int GetIsInvisibleTransition_xpAM(object oObject);
```

Get the Invisible Transition flag for oObject. Work for Door.

Parameters

- `oObject` - The object to get the Invisible Transition flag for.

Return Value

The value of the Invisible Transition flag for the object (TRUE if enabled, FALSE if disabled).

SetIsPartyTransition_xpAM

```
void SetIsPartyTransition_xpAM(object oObject, int bPartyTransition);
```

Set the Party Transition flag for oObject. Work for Door or Trigger.

Parameters

- `oObject` - The object to set the Party Transition flag for.
- `bPartyTransition` - The flag to enable (TRUE) or disable (FALSE) Party Transition.

GetIsPartyTransition_xpAM

```
int GetIsPartyTransition_xpAM(object oObject);
```

Get the Party Transition flag for oObject. Work for Door or Trigger.

Parameters

- `oObject` - The object to get the Party Transition flag for.

Return Value

The value of the Party Transition flag for the object (TRUE if enabled, FALSE if disabled).

SetTransitionLinkObjectType_xpAM

```
void SetTransitionLinkObjectType_xpAM(object oObject, int iType);
```

Set the Transition Link Object Type of oObject to iType. iType must be a [TRANSITION_TYPE *](#) constant. Work for Door or Trigger.

Parameters

- `oObject` - The object to set the Transition Link Object Type for.
- `iType` - The Transition Link Object Type as a [TRANSITION_TYPE *](#) constant.

GetTransitionLinkObjectType_xpAM

```
int GetTransitionLinkObjectType_xpAM(object oObject);
```

Get the [TRANSITION_TYPE *](#) used for oObject. Work for Door or Trigger.

Parameters

- `oObject` - The object to get the Transition Link Object Type for.

Return Value

The Transition Link Object Type as a [TRANSITION_TYPE *](#) constant.

SetLoadScreenID_xpAM

```
void SetLoadScreenID_xpAM(object oObject, int iIDScreen);
```

Set the LoadScreenID for oObject to iIDScreen. Work for Area, Door, Trigger.

Parameters

- `oObject` - The object to set the LoadScreenID for.
 - `iIDScreen` - The LoadScreenID to set.
-

GetLoadScreenID_xpAM

```
int GetLoadScreenID_xpAM(object oObject);
```

Get the LoadScreenID for oObject. Work for Area, Door, Trigger.

Parameters

- `oObject` - The object to get the LoadScreenID for.

Return Value

The LoadScreenID of the object.

SetTransitionDestination_xpAM

```
void SetTransitionDestination_xpAM(object oObject, string sTransitionDest);
```

Set the Transition Destination (object Tag) for oObject. Work for Door, Trigger.

Parameters

- `oObject` - The object to set the Transition Destination for.
 - `sTransitionDest` - The Transition Destination as an object tag.
-

GetTransitionDestination_xpAM

```
string GetTransitionDestination_xpAM(object oObject);
```

Get the Transition Destination of oObject. Work for Door, Trigger.

Parameters

- `oObject` - The object to get the Transition Destination for.

Return Value

The Transition Destination as an object tag.

Saves

SetMagicResistance_xpAM

```
void SetMagicResistance_xpAM(object oObject, int iValue, int bReduc = FALSE);
```

Set the Magic Resistance for oObject. If bReduc is set to TRUE, you will set the Reduction of this Magic Resistance. Note: If you set bReduc to TRUE, and set an negative iValue, it will increase the magicResistance. Work for Creature.

Parameters

- `oObject` - The object to set the Magic Resistance for.
 - `iValue` - The Magic Resistance value to set.
 - `bReduc` - The flag to indicate if the value is a reduction (TRUE) or not (FALSE).
-

GetMagicResistance_xpAM

```
int GetMagicResistance_xpAM(object oObject, int bReduc = FALSE);
```

Get the Magic Resistance for oObject. If bReduc is set to TRUE, you will get the Reduction value of this Magic Resistance.

Parameters

- `oObject` - The object to get the Magic Resistance for.
- `bReduc` - The flag to indicate if the value is a reduction (TRUE) or not (FALSE).

Return Value

The Magic Resistance value or the reduction value if bReduc is TRUE.

SetReflexSavingThrow_xpAM

```
void SetReflexSavingThrow_xpAM(object oObject, int iSaving);
```

Set the Reflex Saving Throw on a creature, door, or placeable.

Parameters

- `oObject` - The object to set the Reflex Saving Throw for.
 - `iSaving` - The Reflex Saving Throw value to set.
-

SetFortitudeSavingThrow_xpAM

```
void SetFortitudeSavingThrow_xpAM(object oObject, int iSaving);
```

Set the Fortitude Saving Throw on a creature, door, or placeable.

Parameters

- `oObject` - The object to set the Fortitude Saving Throw for.
 - `iSaving` - The Fortitude Saving Throw value to set.
-

SetWillSavingThrow_xpAM

```
void SetWillSavingThrow_xpAM(object oObject, int iSaving);
```

Set the Will Saving Throw on a creature, door, or placeable.

Parameters

- `oObject` - The object to set the Will Saving Throw for.
 - `iSaving` - The Will Saving Throw value to set.
-

Faction

GetFactionID_xpAM

```
int GetFactionID_xpAM(object oObject);
```

Return the FactionID of oObject. Work for Creature, Door, Placeable, Trigger.

Parameters

- `oObject` - The object to get the FactionID for.

Return Value

The FactionID of the object.

GetIsBaseFaction_xpAM

```
int GetIsBaseFaction_xpAM(int idFaction);
```

Return TRUE if the idFaction is one of the preconfiguredFaction.

Parameters

- `idFaction` - The FactionID to check.

Return Value

TRUE if the FactionID is a preconfigured faction, FALSE otherwise.

Visual

SetAppearanceSEF_xpAM

```
void SetAppearanceSEF_xpAM(object oObject, string sSEF);
```

Set sSEF visual effect to oObject. Work for Creature, Item, Placeable, PlacedEffect.

Parameters

- `oObject` - The object to set the visual effect for.
 - `sSEF` - The visual effect to set.
-

GetAppearanceSEF_xpAM

```
string GetAppearanceSEF_xpAM(object oObject);
```

Get the current visual effect applied to oObject. Work for Creature, Item, Placeable, PlacedEffect.

Parameters

- `oObject` - The object to get the visual effect for.

Return Value

The current visual effect applied to the object.

SetColor_xpAM

```
void SetColor_xpAM(object oObject, int iColor, string sRGB);
```


Set the iColor color of oObject to sRGB. oObject: The object to change. Door or Placeable. iColor: 1, 2 or 3. The color to change. sRGB: color in #RRGGBB format.

Parameters

- `oObject` - The object to set the color for.
 - `iColor` - The color to change (1, 2, or 3).
 - `sRGB` - The color in #RRGGBB format.
-

GetColor_xpAM

```
string GetColor_xpAM(object oObject, int iColor);
```

Return the "iColor" color of oObject. In #RRGGBB format. oObject: a Door or Placeable object. iColor: 1, 2 or 3. The color to get.

Parameters

- `oObject` - The object to get the color for.
- `iColor` - The color to get (1, 2, or 3).

Return Value

The color in #RRGGBB format.

Position and facing

SetFacing_xpAM

```
void SetFacing_xpAM(object oObject, float fFacing);
```

Cause the oObject to immediately face fFacing. fFacing is expressed as anticlockwise degrees from Due East.

Parameters

- `oObject` - The object to set the facing for.
 - `fFacing` - The facing in anticlockwise degrees from Due East.
-

SetPosition_xpAM

```
void SetPosition_xpAM(object oObject, float fX, float fY, float fZ);
```

Set the position of oObject.

Parameters

- `oObject` - The object to set the position for.
 - `fX`, `fY`, `fZ` - The coordinates of the new position.
-

SetPositionAndFacing_xpAM

```
void SetPositionAndFacing_xpAM(object oObject, float fX, float fY, float fZ, float fFacing);
```

Set the position and orientation of `oObject`.

Parameters

- `oObject` - The object to set the position and facing for.
 - `fX`, `fY`, `fZ` - The coordinates of the new position.
 - `fFacing` - The facing in anticlockwise degrees from Due East.
-

Music

MusicBackgroundDayForPC_xpAM

```
void MusicBackgroundDayForPC_xpAM(object oPC, int nTrack);
```

Change the current `MusicBackgroundDay` to `nTrack` for `oPC` in its current area.

Parameters

- `oPC` - The PC to change the `MusicBackgroundDay` for.
 - `nTrack` - The new `MusicBackgroundDay` track.
-

MusicBackgroundNightForPC_xpAM

```
void MusicBackgroundNightForPC_xpAM(object oPC, int nTrack);
```

Change the current `MusicBackgroundNight` to `nTrack` for `oPC` in its current area.

Parameters

- `oPC` - The PC to change the `MusicBackgroundNight` for.
 - `nTrack` - The new `MusicBackgroundNight` track.
-

MusicBattleForPC_xpAM

```
void MusicBattleForPC_xpAM(object oPC, int nTrack);
```

Change the current `MusicBattle` to `nTrack` for `oPC` in its current area.

Parameters

- `oPC` - The PC to change the MusicBattle for.
 - `nTrack` - The new MusicBattle track.
-

MusicBackgroundRunForPC_xpAM

```
void MusicBackgroundRunForPC_xpAM(object oPC, int bStart);
```

Start or Stop the current MusicBackground for oPC.

Parameters

- `oPC` - The PC to start or stop the MusicBackground for.
 - `bStart` - The flag to start (TRUE) or stop (FALSE) the MusicBackground.
-

MusicBattleRunForPC_xpAM

```
void MusicBattleRunForPC_xpAM(object oPC, int bStart);
```

Start or Stop the current MusicBattle for oPC.

Parameters

- `oPC` - The PC to start or stop the MusicBattle for.
 - `bStart` - The flag to start (TRUE) or stop (FALSE) the MusicBattle.
-

Refresh, View & Collision

RefreshCreatureForPC_xpAM

```
void RefreshCreatureForPC_xpAM(object oCreatureToRefresh, object oPCMustView);
```

Refresh the visual of oCreatureToRefresh for oPCMustView.

Parameters

- `oCreatureToRefresh` - The creature to refresh the visual for.
 - `oPCMustView` - The PC that must view the refreshed visual.
-

RefreshCreature_xpAM

```
void RefreshCreature_xpAM(object oCreatureToRefresh);
```

Refresh the visual of oCreatureToRefresh for every PC.

Parameters

- `oCreatureToRefresh` - The creature to refresh the visual for.
-

RefreshEquippedItemForPC_xpAM

```
void RefreshEquippedItemForPC_xpAM(int iSlotToRefresh, object oCreatureToRefresh, object oMustView);
```

Refresh the visual of an equipped item of `oCreatureToRefresh` for `oPCMustView`.
`iSlotToRefresh`: `INVENTORY_SLOT_*` constant.

Parameters

- `iSlotToRefresh` - The inventory slot to refresh the equipped item for (`INVENTORY_SLOT_*` constant).
 - `oCreatureToRefresh` - The creature to refresh the equipped item visual for.
 - `oMustView` - The PC that must view the refreshed visual.
-

RefreshEquippedItem_xpAM

```
void RefreshEquippedItem_xpAM(int iSlotToRefresh, object oCreatureToRefresh);
```

Refresh the visual of an equipped item of `oCreatureToRefresh` for every PC.
`iSlotToRefresh`: `INVENTORY_SLOT_*` constant.

Parameters

- `iSlotToRefresh` - The inventory slot to refresh the equipped item for (`INVENTORY_SLOT_*` constant).
 - `oCreatureToRefresh` - The creature to refresh the equipped item visual for.
-

RefreshObjectForPC_xpAM

```
void RefreshObjectForPC_xpAM(object oObjectToRefresh, object oPC);
```

Refresh the visual of the object `oObjectToRefresh` for `oPC`.

Parameters

- `oObjectToRefresh` - The object to refresh the visual for.
 - `oPC` - The PC that must view the refreshed visual.
-

RefreshObject_xpAM

```
void RefreshObject_xpAM(object oObjectToRefresh);
```

Refresh the visual of the object oObjectToRefresh for every concerned.

Parameters

- oObjectToRefresh - The object to refresh the visual for.
-

ShowObject_xpAM

```
void ShowObject_xpAM(object oObjectToShow);
```

Show the visual of the object oObjectToShow.

Parameters

- oObjectToShow - The object to show the visual for.
-

ShowObjectForPC_xpAM

```
void ShowObjectForPC_xpAM(object oObjectToShow, object oPC);
```

Show the visual of the object oObjectToShow for oPC.

Parameters

- oObjectToShow - The object to show the visual for.
 - oPC - The PC that must view the shown visual.
-

MaskObject_xpAM

```
void MaskObject_xpAM(object oObjectToMask);
```

Hide the visual of the object oObjectToMask for every concerned.

Parameters

- oObjectToMask - The object to hide the visual for.
-

MaskObjectForPC_xpAM

```
void MaskObjectForPC_xpAM(object oObjectToMask, object oPC);
```

Hide the visual of the object oObjectToMask for oPC.

Parameters

- `oObjectToMask` - The object to hide the visual for.
 - `oPC` - The PC that must view the hidden visual.
-

SetObjectDynamicCollision_xpAM

```
void SetObjectDynamicCollision_xpAM(object oPlaceable, int bDynamicCollision);
```

Enable or Disable DynamicCollision of oPlaceable.

Parameters

- `oPlaceable` - The placeable object to enable or disable DynamicCollision for.
 - `bDynamicCollision` - The flag to enable (TRUE) or disable (FALSE) DynamicCollision.
-

GetObjectDynamicCollision_xpAM

```
int GetObjectDynamicCollision_xpAM(object oPlaceable);
```

Get status of DynamicCollision of oPlaceable.

Parameters

- `oPlaceable` - The placeable object to get the DynamicCollision status for.

Return Value

The status of DynamicCollision for the object (TRUE if enabled, FALSE if disabled).

Constants

Light Color Types (LIGHT_COLOR_TYPE_*)

Name	Value	Description
LIGHT_COLOR_TYPE_AMBIANT	"Ambiant"	
LIGHT_COLOR_TYPE_DIFFUSE	"Diffuse"	
LIGHT_COLOR_TYPE_SPECULAR	"Specular"	

Transition Types (TRANSITION_TYPE_*)

Name	Value	Description
TRANSITION_TYPE_NONE	0	
TRANSITION_TYPE_DOOR	1	
TRANSITION_TYPE_WAYPOINT	2	

Flicker Types (FLICKER_TYPE_*)

Name	Value	Description
FLICKER_TYPE_LINEAR	0	
FLICKER_TYPE_BOUNCE	1	
FLICKER_TYPE_JUMPY	2	

Armor Model Parts (ARMOR_MODEL_PART_*)

Name	Value	Description
ARMOR_MODEL_PART_HELM	0	Already existing in base nwscript
ARMOR_MODEL_PART_GLOVES	1	Already existing in base nwscript
ARMOR_MODEL_PART_BOOTS	2	Already existing in base nwscript
ARMOR_MODEL_PART_BELT	3	Already existing in base nwscript
ARMOR_MODEL_PART_CLOAK	4	Already existing in base nwscript
ARMOR_MODEL_PART_ARMOR	5	Added by xp_aspectManager

Trigger Types (TRIGGER_TYPE_*)

--	--	--

Name	Value	Description
TRIGGER_TYPE_GENERAL	0	
TRIGGER_TYPE_TRANSITION	1	
TRIGGER_TYPE_TRAP	2	

--

xp_aspectManager_creature.nss

- [Global Note](#)
- [SetCreatureAge_xpAM](#)
- [GetCreatureAge_xpAM](#)
- [SetCreatureSoundSet_xpAM](#)
- [GetCreatureSoundSet_xpAM](#)
- [SetCreatureSpecificAC_xpAM](#)
- [GetCreatureSpecificAC_xpAM](#)
- [SetCreatureTattoos_xpAM](#)
- [GetCreatureTattoos_xpAM](#)
- [Creature Appearance](#)
 - [SetCreatureVisualGender_xpAM](#)
 - [GetCreatureVisualGender_xpAM](#)
 - [SetCreatureHead_xpAM](#)
 - [GetCreatureHead_xpAM](#)
 - [SetCreatureHair_xpAM](#)
 - [GetCreatureHair_xpAM](#)
 - [SetCreatureFacialHair_xpAM](#)
 - [GetCreatureFacialHair_xpAM](#)
 - [SetCreatureTail_xpAM](#)
 - [GetCreatureTail_xpAM](#)
 - [SetCreatureWings_xpAM](#)
 - [GetCreatureWings_xpAM](#)
 - [SetCreatureColor_xpAM](#)
 - [GetCreatureColor_xpAM](#)
 - [SetCreaturePersistScale_xpAM](#)
 - [GetCreaturePersistScale_xpAM](#)
- [Creature Visual](#)
 - [SetCreatureModelPartMask_xpAM](#)
 - [GetCreatureModelPartMask_xpAM](#)
 - [SetCreatureVisualVariation_xpAM](#)
 - [GetCreatureVisualVariation_xpAM](#)
 - [SetCreatureVisualCategory_xpAM](#)
 - [GetCreatureVisualCategory_xpAM](#)
 - [SetCreatureVisualColor_xpAM](#)
 - [GetCreatureVisualColor_xpAM](#)
 - [Visual ModelPart](#)
 - [SetCreatureVisualModelPartVariation_xpAM](#)
 - [GetCreatureVisualModelPartVariation_xpAM](#)
 - [SetCreatureVisualModelPartCategory_xpAM](#)
 - [GetCreatureVisualModelPartCategory_xpAM](#)
 - [SetCreatureVisualModelPartColor_xpAM](#)
 - [GetCreatureVisualModelPartColor_xpAM](#)
 - [Visual ModelPiece](#)
 - [SetCreatureVisualModelPieceVariation_xpAM](#)
 - [GetCreatureVisualModelPieceVariation_xpAM](#)
 - [SetCreatureVisualModelPieceColor_xpAM](#)
 - [GetCreatureVisualModelPieceColor_xpAM](#)
 - [SetCreatureVisualModelPieceIsUVScroll_xpAM](#)
 - [GetCreatureVisualModelPieceIsUVScroll_xpAM](#)

- [SetCreatureVisualModelPieceUScroll_xpAM](#)
- [GetCreatureVisualModelPieceUScroll_xpAM](#)
- [SetCreatureVisualModelPieceVScroll_xpAM](#)
- [GetCreatureVisualModelPieceVScroll_xpAM](#)
- **Constants**
 - [Creature Color Constants \(CREATURE_COLOR *\)](#).
 - [Armor Class Type Constants \(AC TYPE *\)](#).
 - [Scale Axis Constants \(SCALE *\)](#).

Global Note

For some parts, Creatures use a set of two values. One is used for the current usage (e.g., the current Hair shown), while the other one represents the "persistent" value, which will be the only one exported/saved.

To allow you to choose which one to change, the relevant function has a `bPersist` parameter. Set it to `FALSE` to change the "current" value, and to `TRUE` to change the persisted one.

Quick example:

- My oPC has the head value set to 4. Therefore, both the persistent and current head values are set to 4.
 - I call `SetCreatureHead_xpAM(oPC, 5, FALSE)` . => The head of PC changes to 5.
 - The PC leaves the server and rejoins => the head will be back to 4.
 - I call `SetCreatureHead_xpAM(oPC, 5, TRUE)` => nothing visible happens.
 - The PC leaves the server and rejoins again => the change is saved, new head is 5.

Note: Function without the `bPersist` parameter will impact "unique" value, that will affect the current AND be persisted.

SetCreatureAge_xpAM

```
void SetCreatureAge_xpAM(
    object oCreature,
    int iAge,
    int bPersist = FALSE
);
```

Set the Age of oCreature

Parameters

- `oCreature` - The creature to set the age for
- `iAge` - The age to set
- `bPersist` - Set to `TRUE` to change the persisted value, `FALSE` to change the current value.

GetCreatureAge_xpAM

```
int GetCreatureAge_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Age of oCreature

Parameters

- `oCreature` - The creature to get the age for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The age of the creature.

SetCreatureSoundSet_xpAM

```
void SetCreatureSoundSet_xpAM(  
    object oCreature,  
    int iSoundSet  
);
```

Set the SoundSet of oCreature

Parameters

- `oCreature` - The creature to set the sound set for
 - `iSoundSet` - The sound set to use
-

GetCreatureSoundSet_xpAM

```
int GetCreatureSoundSet_xpAM(  
    object oCreature  
);
```

Get the SoundSet of oCreature

Parameters

- `oCreature` - The creature to get the sound set for

Return Value

The sound set of the creature.

SetCreatureSpecificAC_xpAM

```
void SetCreatureSpecificAC_xpAM(  
    object oCreature,  
    int iACType,  
    int iValue,  
    int bReduce = FALSE  
);
```

Set the Specific Armor Class (AC) of oCreature

Parameters

- `oCreature` - The creature to set the AC for
- `iACType` - ([AC_TYPE *](#)) constant representing the AC to be set.
- `iValue` - a number between -128 and +127 that determines the new value of the AC
- `bReduce` - Set to TRUE to decrease the AC, FALSE to increase it
 - All AC_TYPE except BASE, ARMORVAL, SHIELDVAL have a "ReduceValue".
 - Positive `iValue` will increase the chosen AC if `bReduce=FALSE`, and decrease it if `bReduce=TRUE`.
 - Negative values will decrease the chosen AC if `bReduce=FALSE`, but increase it if `bReduce=TRUE`.

Note

If you set a specific AC, it will be overwritten by other factors that also set the same AC. For example, setting AC_TYPE_ARMORVAL_SEPT to 18 will be overwritten when the oCreature equips armor. Another example: when a creature receives Mage Armor, the AC_TYPE_ARMOR_SEPT will be overwritten but NOT the reduced AC_TYPE_ARMOR_SEPT.

GetCreatureSpecificAC_xpAM

```
int GetCreatureSpecificAC_xpAM(  
    object oCreature,  
    int iACType,  
    int bReduceVal = FALSE  
);
```

Get the Specific Armor Class (AC) of oCreature

Parameters

- `oCreature` - The creature to get the AC for
- `iACType` - ([AC_TYPE *](#)) constant representing the AC to be get.
- `bReduceVal` - Set to TRUE to get the reduced value, FALSE to get the normal value All AC_TYPE except BASE, ARMORVAL, SHIELDVAL have a "ReduceValue".

Return Value

The specific armor class (AC) of the creature.

SetCreatureTattoos_xpAM

```
void SetCreatureTattoos_xpAM(  
    object oCreature,  
    int iValue  
);
```

Not working for tattoo but can be used to store a 16bits value.

Parameters

- `oCreature` - The creature to store the value for
 - `iValue` - The 16bits value to store
-

GetCreatureTattoos_xpAM

```
int GetCreatureTattoos_xpAM(  
    object oCreature  
);
```

Not working for tattoo but can be used to get a stored 16bits value.

Parameters

- `oCreature` - The creature to get the value for

Return Value

The stored 16bits value.

Creature Appearance

SetCreatureVisualGender_xpAM

```
void SetCreatureVisualGender_xpAM(  
    object oCreature,  
    int iVisualGender  
);
```

Needed if you change the Gender of an creature

Parameters

- `oCreature` - The creature to set the visual gender for
 - `iVisualGender` - The visual gender to set
-

GetCreatureVisualGender_xpAM

```
int GetCreatureVisualGender_xpAM(  
    object oCreature  
);
```

Get the visual gender of a creature

Parameters

- `oCreature` - The creature to get the visual gender for

Return Value

The visual gender of the creature.

SetCreatureHead_xpAM

```
void SetCreatureHead_xpAM(  
    object oCreature,  
    int iVariation,  
    int bPersist = FALSE  
);
```

Change the Head of oCreature.

Parameters

- `oCreature` - The creature to change the head for
 - `iVariation` - The head variation to use
 - `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureHead_xpAM

```
int GetCreatureHead_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Head of oCreature

Parameters

- `oCreature` - The creature to get the head for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The head variation of the creature.

SetCreatureHair_xpAM

```
void SetCreatureHair_xpAM(  
    object oCreature,  
    int iVariation,  
    int bPersist = FALSE  
);
```

Change the Hair of oCreature.

Parameters

- `oCreature` - The creature to change the hair for
 - `iVariation` - The hair variation to use
 - `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureHair_xpAM

```
int GetCreatureHair_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Hair of oCreature

Parameters

- `oCreature` - The creature to get the hair for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The hair variation of the creature.

SetCreatureFacialHair_xpAM

```
void SetCreatureFacialHair_xpAM(  
    object oCreature,  
    int iVariation,  
    int bPersist = FALSE  
);
```

Change the Facial Hair of oCreature.

Parameters

- `oCreature` - The creature to change the facial hair for
 - `iVariation` - The facial hair variation to use
 - `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureFacialHair_xpAM

```
int GetCreatureFacialHair_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Facial Hair of oCreature

Parameters

- `oCreature` - The creature to get the facial hair for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The facial hair variation of the creature.

SetCreatureTail_xpAM

```
void SetCreatureTail_xpAM(  
    object oCreature,  
    int iVariation,  
    int bPersist = FALSE  
);
```

Change the Tail of oCreature.

Parameters

- `oCreature` - The creature to change the tail for
- `iVariation` - The tail variation to use

- `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureTail_xpAM

```
int GetCreatureTail_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Tail of oCreature

Parameters

- `oCreature` - The creature to get the tail for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The tail variation of the creature.

SetCreatureWings_xpAM

```
void SetCreatureWings_xpAM(  
    object oCreature,  
    int iVariation,  
    int bPersist = FALSE  
);
```

Change the Wings of oCreature.

Parameters

- `oCreature` - The creature to change the wings for
 - `iVariation` - The wings variation to use
 - `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureWings_xpAM

```
int GetCreatureWings_xpAM(  
    object oCreature,  
    int bPersist = FALSE  
);
```

Get the Wings of oCreature

Parameters

- `oCreature` - The creature to get the wings for
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The wings variation of the creature.

SetCreatureColor_xpAM

```
void SetCreatureColor_xpAM(  
    object oCreature,  
    string sCreatureColor,  
    string sRGB,  
    int bPersist = FALSE  
);
```

Change the choosen color of oCreature.

Parameters

- `oCreature` - The creature to change the color for
 - `sCreatureColor` - [CREATURE_COLOR](#) * constant, the choosen color
 - `sRGB` - a RGB string value, on the "#FFFFFF" (or #FFFFFFF) format
 - `bPersist` - Set to TRUE to change the persisted value, FALSE to change the current value.
-

GetCreatureColor_xpAM

```
string GetCreatureColor_xpAM(  
    object oCreature,  
    string sCreatureColor,  
    int bPersist = FALSE  
);
```

Get the choosen color of oCreature.

Parameters

- `oCreature` - The creature to get the color for
- `sCreatureColor` - [CREATURE_COLOR](#) * constant, the choosen color
- `bPersist` - Set to TRUE to get the persisted value, FALSE to get the current value.

Return Value

The RGB string value of the chosen color.

SetCreaturePersistScale_xpAM

```
void SetCreaturePersistScale_xpAM(  
    object oCreature,  
    int nAxis,  
    float fScale  
);
```

Change the Persisted scale of oCreature based on specific nAxis.

Parameters

- oCreature - The creature to change the scale for
- nAxis - [SCALE *](#) constant, the scale axis to set.
- fScale - The scale value to use

GetCreaturePersistScale_xpAM

```
float GetCreaturePersistScale_xpAM(  
    object oCreature,  
    int nAxis  
);
```

Get the Persisted scale of the object based on which axis is looked up.

Parameters

- oCreature - The creature to get the scale for
- nAxis - [SCALE *](#) constant, the scale axis to get.

Return Value

The scale value of the creature on the specified axis.

Creature Visual

SetCreatureModelPartMask_xpAM

```
void SetCreatureModelPartMask_xpAM(  
    object oCreature,  
    int iMask  
);
```

Change the iModelPart mask, allowing you to show or hide model part on the creature visual (so without armor)

Parameters

- `oCreature` - The creature to change the model part mask for
- `iMask` - The model part mask to use

Note

`iMask` is bitmask based with : 0x1: Helm, 0x2: Gloves, 0x4: Boots, 0x8: Belt, 0x10: Cloak

GetCreatureModelPartMask_xpAM

```
int GetCreatureModelPartMask_xpAM(  
    object oCreature  
);
```

Get the `iModelPart` mask, tell you to which model part are shown or hidden on the creature visual (so without armor)

Parameters

- `oCreature` - The creature to get the model part mask for

Return Value

The model part mask of the creature.

Note

Return Value is bitmask based with : 0x1: Helm, 0x2: Gloves, 0x4: Boots, 0x8: Belt, 0x10: Cloak

SetCreatureVisualVariation_xpAM

```
void SetCreatureVisualVariation_xpAM(  
    object oCreature,  
    int iVariation  
);
```

Set the `iVariation` for the `oCreature` model (without armor)

Parameters

- `oCreature` - The creature to change the model variation for
 - `iVariation` - The model variation to use
-

GetCreatureVisualVariation_xpAM

```
int GetCreatureVisualVariation_xpAM(  
    object oCreature  
);
```

Get the iVariation for the oCreature model (without armor)

Parameters

- `oCreature` - The creature to get the model variation for

Return Value

The model variation of the creature.

SetCreatureVisualCategory_xpAM

```
void SetCreatureVisualCategory_xpAM(  
    object oCreature,  
    int iCategory  
);
```

Set the Category (armorvisualdata.2da) for the oCreature model (without armor)

Parameters

- `oCreature` - The creature to change the model category for
 - `iCategory` - The model category to use
-

GetCreatureVisualCategory_xpAM

```
int GetCreatureVisualCategory_xpAM(  
    object oCreature  
);
```

Get the Category (armorvisualdata.2da) for the oCreature model (without armor)

Parameters

- `oCreature` - The creature to get the model category for

Return Value

The model category of the creature.

SetCreatureVisualColor_xpAM

```
void SetCreatureVisualColor_xpAM(  
    object oCreature,  
    int iColor,  
    string sRGB  
);
```

Set the visual Color for oCreature (Seems to be persisted but not used)

Parameters

- `oCreature` - The creature to change the visual color for
- `iColor` - The color to use
- `sRGB` - a RGB string value, on the "#FFFFFF" (or #FFFFFF) format

GetCreatureVisualColor_xpAM

```
string GetCreatureVisualColor_xpAM(  
    object oCreature,  
    int iColor  
);
```

Get the visual Color for oCreature (Seems to be persisted but not used)

Parameters

- `oCreature` - The creature to get the visual color for
- `iColor` - The color to get

Return Value

The RGB string value of the visual color.

Visual ModelPart

SetCreatureVisualModelPartVariation_xpAM

```
void SetCreatureVisualModelPartVariation_xpAM(  
    object oCreature,  
    int iModelPart,  
    int iVariation  
);
```

Set the ModelPart Variation for oCreature Visual.

Parameters

- `oCreature` - The creature to change the model part variation for
- `iModelPart` - `ARMOR_MODEL_PART_*` constant

- `iVariation` - The model part variation to use
-

GetCreatureVisualModelPartVariation_xpAM

```
int GetCreatureVisualModelPartVariation_xpAM(  
    object oCreature,  
    int iModelPart  
);
```

Get the ModelPart Variation for oCreature Visual.

Parameters

- `oCreature` - The creature to get the model part variation for
- `iModelPart` - ARMOR_MODEL_PART_* constant

Return Value

The model part variation of the creature.

SetCreatureVisualModelPartCategory_xpAM

```
void SetCreatureVisualModelPartCategory_xpAM(  
    object oCreature,  
    int iModelPart,  
    int iCategory  
);
```

Set the ModelPart Visual Category for oCreature Visual (armorvisualdata.2DA ID).

Parameters

- `oCreature` - The creature to change the model part category for
 - `iModelPart` - ARMOR_MODEL_PART_* constant
 - `iCategory` - The model part category to use
-

GetCreatureVisualModelPartCategory_xpAM

```
int GetCreatureVisualModelPartCategory_xpAM(  
    object oCreature,  
    int iModelPart  
);
```

Get the ModelPart Visual Category for oCreature Visual (armorvisualdata.2DA ID).

Parameters

- `oCreature` - The creature to get the model part category for
- `iModelPart` - `ARMOR_MODEL_PART_*` constant

Return Value

The model part category of the creature.

SetCreatureVisualModelPartColor_xpAM

```
void SetCreatureVisualModelPartColor_xpAM(  
    object oCreature,  
    int iModelPart,  
    int iColor,  
    string sRGB  
);
```

Set the Color of the `iModelPart` of `oCreature` Visual

Parameters

- `oCreature` - The creature to change the model part color for
 - `iModelPart` - `ARMOR_MODEL_PART_*` constant
 - `iColor` - 1, 2 or 3
 - `sRGB` - a RGB string value, on the "#FFFFFF" (or #FFFFFF) format
-

GetCreatureVisualModelPartColor_xpAM

```
string GetCreatureVisualModelPartColor_xpAM(  
    object oCreature,  
    int iModelPart,  
    int iColor  
);
```

Get the Color of the `iModelPart` of `oCreature` Visual

Parameters

- `oCreature` - The creature to get the model part color for
- `iModelPart` - `ARMOR_MODEL_PART_*` constant
- `iColor` - 1, 2 or 3

Return Value

The RGB string value of the model part color.

Visual ModelPiece

SetCreatureVisualModelPieceVariation_xpAM

```
void SetCreatureVisualModelPieceVariation_xpAM(  
    object oCreature,  
    int iModelPiece,  
    int iVariation  
);
```

Set the ModelPiece Variation for oCreature Visual. Use ARMOR_MODEL_PIECE_* constant for iModelPiece

Parameters

- `oCreature` - The creature to change the model piece variation for
 - `iModelPiece` - ARMOR_MODEL_PIECE_* constant
 - `iVariation` - The model piece variation to use
-

GetCreatureVisualModelPieceVariation_xpAM

```
int GetCreatureVisualModelPieceVariation_xpAM(  
    object oCreature,  
    int iModelPiece  
);
```

Get the ModelPiece Variation for oCreature Visual. Use ARMOR_MODEL_PIECE_* constant for iModelPiece

Parameters

- `oCreature` - The creature to get the model piece variation for
- `iModelPiece` - ARMOR_MODEL_PIECE_* constant

Return Value

The model piece variation of the creature.

SetCreatureVisualModelPieceColor_xpAM

```
void SetCreatureVisualModelPieceColor_xpAM(  
    object oCreature,  
    int iModelPiece,  
    int iColor,  
    string sRGB  
);
```

Set the Color of the iModelPiece of oCreature Visual

Parameters

- `oCreature` - The creature to change the model piece color for
 - `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant
 - `iColor` - 1, 2 or 3
 - `sRGB` - a RGB string value, on the "#FFFFFF" (or #FFFFFFF) format
-

GetCreatureVisualModelPieceColor_xpAM

```
string GetCreatureVisualModelPieceColor_xpAM(  
    object oCreature,  
    int iModelPiece,  
    int iColor  
);
```

Get the Color of the `iModelPiece` of `oCreature` Visual

Parameters

- `oCreature` - The creature to get the model piece color for
- `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant
- `iColor` - 1, 2 or 3

Return Value

The RGB string value of the model piece color.

SetCreatureVisualModelPieceIsUVScroll_xpAM

```
void SetCreatureVisualModelPieceIsUVScroll_xpAM(  
    object oCreature,  
    int iModelPiece,  
    int bActivate  
);
```

Enable or Disable the UVScroll of the `iModelPiece` of `oCreature` Visual

Parameters

- `oCreature` - The creature to enable or disable the UVScroll for
 - `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant
 - `bActivate` - Set to TRUE to enable the UVScroll, FALSE to disable it
-

GetCreatureVisualModelPieceIsUVScroll_xpAM

```
int GetCreatureVisualModelPieceIsUVScroll_xpAM(  
    object oCreature,
```

```
int iModelPiece  
);
```

Get the activation status for the UVScroll of the iModelPiece of oCreature Visual

Parameters

- `oCreature` - The creature to get the UVScroll activation status for
- `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant

Return Value

The activation status of the UVScroll for the model piece.

SetCreatureVisualModelPieceUScroll_xpAM

```
void SetCreatureVisualModelPieceUScroll_xpAM(  
    object oCreature,  
    int iModelPiece,  
    float fValue  
);
```

Set the UScroll speed for the iModelPiece of oCreature Visual

Parameters

- `oCreature` - The creature to change the UScroll speed for
 - `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant
 - `fValue` - The UScroll speed to use
-

GetCreatureVisualModelPieceUScroll_xpAM

```
float GetCreatureVisualModelPieceUScroll_xpAM(  
    object oCreature,  
    int iModelPiece  
);
```

Get the UScroll speed for the iModelPiece of oCreature Visual

Parameters

- `oCreature` - The creature to get the UScroll speed for
- `iModelPiece` - `ARMOR_MODEL_PIECE_*` constant

Return Value

The UScroll speed of the model piece.

SetCreatureVisualModelPieceVScroll_xpAM

```
void SetCreatureVisualModelPieceVScroll_xpAM(  
    object oCreature,  
    int iModelPiece,  
    float fValue  
);
```

Set the VScroll speed for the iModelPiece of oCreature Visual

Parameters

- oCreature - The creature to change the VScroll speed for
- iModelPiece - ARMOR_MODEL_PIECE_* constant
- fValue - The VScroll speed to use

GetCreatureVisualModelPieceVScroll_xpAM

```
float GetCreatureVisualModelPieceVScroll_xpAM(  
    object oCreature,  
    int iModelPiece  
);
```

Get the VScroll speed for the iModelPiece of oCreature Visual

Parameters

- oCreature - The creature to get the VScroll speed for
- iModelPiece - ARMOR_MODEL_PIECE_* constant

Return Value

The VScroll speed of the model piece.

Constants Creature

Creature Color Constants (CREATURE_COLOR_*)

Name	Value	Description
CREATURE_COLOR_1_SEPT	"Color1"	Creature color 1
CREATURE_COLOR_2_SEPT	"Color2"	Creature color 2
CREATURE_COLOR_3_SEPT	"Color3"	Creature color 3
CREATURE_COLOR_HEAD_SEPT	"ColorHeadSkin"	Head/Skin color
CREATURE_COLOR_EYE_SEPT	"ColorHeadEye"	Eye color

CREATURE_COLOR_BROW_SEPT	"ColorHeadBrow"	Brow color
CREATURE_COLOR_HAIRACCESSORY_SEPT	"ColorHairHacc"	Hair accessory color
CREATURE_COLOR_HAIR1_SEPT	"ColorHair1"	Hair color 1
CREATURE_COLOR_HAIR2_SEPT	"ColorHair2"	Hair color 2

Armor Class Type Constants (AC_TYPE_*)

Name	Value	Description
AC_TYPE_BASE_SEPT	20	Base armor class
AC_TYPE_ARMORVAL_SEPT	21	Armor value
AC_TYPE_SHIELDVAL_SEPT	22	Shield value
AC_TYPE_ARMOR_SEPT	3	Armor
AC_TYPE_DEFLECT_SEPT	5	Deflection
AC_TYPE_SHIELD_SEPT	7	Shield
AC_TYPE_NATURAL_SEPT	9	Natural armor
AC_TYPE_DODGE_SEPT	11	Dodge

Scale Axis Constants (SCALE_*)

Name	Value	Description
SCALE_X	0	X-axis scale
SCALE_Y	1	Y-axis scale
SCALE_Z	2	Z-axis scale

--

xp_aspectManager_area.nss

- [Area Settings](#)
 - [SetAreaFlags_xpAM](#)
 - [GetAreaFlags_xpAM](#)
 - [SetAreaNoRest_xpAM](#)
 - [GetAreaNoRest_xpAM](#)
 - [SetAreaPVPMode_xpAM](#)
 - [GetAreaPVPMode_xpAM](#)
 - [SetAreaModSpotCheck_xpAM](#)
 - [GetAreaModSpotCheck_xpAM](#)
 - [SetAreaModListenCheck_xpAM](#)
 - [GetAreaModListenCheck_xpAM](#)
 - [SetAreaSkyRing_xpAM](#)
 - [GetAreaSkyRing_xpAM](#)
 - [SetAreaHasDirectionalLight_xpAM](#)
 - [GetAreaHasDirectionalLight_xpAM](#)
 - [SetAreaIsDirLightCastShadow_xpAM](#)
 - [GetAreaIsDirLightCastShadow_xpAM](#)
- [Day/Night Cycle](#)
 - [SetAreaHasDayNightCycle_xpAM](#)
 - [GetAreaHasDayNightCycle_xpAM](#)
 - [SetAreaIsAlwaysNight_xpAM](#)
 - [GetAreaIsAlwaysNight_xpAM](#)
 - [SetAreaUseDayNightFog_xpAM](#)
 - [GetAreaUseDayNightFog_xpAM](#)
 - [SetAreaDNCCoronaIntensity_xpAM](#)
 - [GetAreaDNCCoronaIntensity_xpAM](#)
 - [SetAreaDNCSHadowIntensity_xpAM](#)
 - [GetAreaDNCSHadowIntensity_xpAM](#)
 - [SetAreaDNCSkyRingColorInterpretation_xpAM](#)
 - [GetAreaDNCSkyRingColorInterpretation_xpAM](#)
 - [SetAreaDNCSkyDomeModel_xpAM](#)
 - [GetAreaDNCSkyDomeModel_xpAM](#)
 - [Bloom](#)
 - [SetAreaDNCBloomBLRadius_xpAM](#)
 - [GetAreaDNCBloomBLRadius_xpAM](#)
 - [SetAreaDNCBloomGLIntensity_xpAM](#)
 - [GetAreaDNCBloomGLIntensity_xpAM](#)
 - [SetAreaDNCBloomHLIntensity_xpAM](#)
 - [GetAreaDNCBloomHLIntensity_xpAM](#)
 - [SetAreaDNCBloomHLThreshold_xpAM](#)
 - [GetAreaDNCBloomHLThreshold_xpAM](#)
 - [SetAreaDNCBloomSCIntensity_xpAM](#)
 - [GetAreaDNCBloomSCIntensity_xpAM](#)
 - [HDR](#)
 - [SetAreaDNCHDRAvg_xpAM](#)
 - [GetAreaDNCHDRAvg_xpAM](#)
 - [SetAreaDNCHDRMax_xpAM](#)
 - [GetAreaDNCHDRMax_xpAM](#)
 - [SetAreaDNCHDRExposure_xpAM](#)

- [GetAreaDNCHDRExposure_xpAM](#)
- [SetAreaDNCHDRHighlight_xpAM](#)
- [GetAreaDNCHDRHighlight_xpAM](#)
- **Cloud**
 - [SetAreaDNCCloudCover_xpAM](#)
 - [GetAreaDNCCloudCover_xpAM](#)
 - [SetAreaDNCCloudMoveX_xpAM](#)
 - [GetAreaDNCCloudMoveX_xpAM](#)
 - [SetAreaDNCCloudMoveY_xpAM](#)
 - [GetAreaDNCCloudMoveY_xpAM](#)
- **Desaturate**
 - [SetAreaDNCDesaturateColor_xpAM](#)
 - [GetAreaDNCDesaturateColor_xpAM](#)
 - [SetAreaDNCDesaturateFactor_xpAM](#)
 - [GetAreaDNCDesaturateFactor_xpAM](#)
- **GroundLight**
 - [SetAreaDNCGroundLightIntensity_xpAM](#)
 - [GetAreaDNCGroundLightIntensity_xpAM](#)
 - [SetAreaDNCGroundLightColor_xpAM](#)
 - [GetAreaDNCGroundLightColor_xpAM](#)
- **SkyLight**
 - [SetAreaDNCSkyLightIntensity_xpAM](#)
 - [GetAreaDNCSkyLightIntensity_xpAM](#)
 - [SetAreaDNCSkyLightColor_xpAM](#)
 - [GetAreaDNCSkyLightColor_xpAM](#)
- **SunMoon**
 - [SetAreaDNCSunMoonIntensity_xpAM](#)
 - [GetAreaDNCSunMoonIntensity_xpAM](#)
 - [SetAreaDNCSunMoonColor_xpAM](#)
 - [GetAreaDNCSunMoonColor_xpAM](#)
 - [SetAreaDNCSunMoonDirX_xpAM](#)
 - [GetAreaDNCSunMoonDirX_xpAM](#)
 - [SetAreaDNCSunMoonDirY_xpAM](#)
 - [GetAreaDNCSunMoonDirY_xpAM](#)
 - [SetAreaDNCSunMoonDirZ_xpAM](#)
 - [GetAreaDNCSunMoonDirZ_xpAM](#)
- **Fog**
 - [SetAreaDNCFogStart_xpAM](#)
 - [GetAreaDNCFogStart_xpAM](#)
 - [SetAreaDNCFogEnd_xpAM](#)
 - [GetAreaDNCFogEnd_xpAM](#)
 - [SetAreaDNCFogFarClip_xpAM](#)
 - [GetAreaDNCFogFarClip_xpAM](#)
 - [SetAreaDNCFogColor_xpAM](#)
 - [GetAreaDNCFogColor_xpAM](#)
- **Other Colors**
 - [SetAreaDNCSkyHorizonColor_xpAM](#)
 - [GetAreaDNCSkyHorizonColor_xpAM](#)
 - [SetAreaDNCSkyZenithColor_xpAM](#)
 - [GetAreaDNCSkyZenithColor_xpAM](#)

- [Constants](#)
 - [PVP Setting_\(PVP_SETTING *\)](#).
 - [Area Flag_\(AREA_FLAG *\)](#).
 - [DayNights_\(DAYNIGHT *\)](#).

Area Settings

SetAreaFlags_xpAM

```
void SetAreaFlags_xpAM(object oArea, int iFlags);
```

Set the AreaFlags (define Exterior, Natural, Subteran or not)

Parameters

- `oArea` - The area to set the flags for
- `iFlags` - The flags to set

Note

`iFlags` can be composed using the [AREA_FLAG *](#) constants. Example : `AREA_FLAG_NATURAL | AREA_FLAG_INTERIOR` will be a natural and interior area.\$

GetAreaFlags_xpAM

```
int GetAreaFlags_xpAM(object oArea);
```

Get the AreaFlags (define Exterior, Natural, Subteran or not)

Parameters

- `oArea` - The area to get the flags for

Return Value

The area flags. Will be composed of [AREA_FLAG *](#) values. For example, an interior and natural area will return a value equal to `AREA_FLAG_NATURAL | AREA_FLAG_INTERIOR`

SetAreaNoRest_xpAM

```
void SetAreaNoRest_xpAM(object oArea, int bNoRest);
```

Enable or disable possibility of Rest in oArea

Parameters

- `oArea` - The area to set the rest restriction for

- `bNoRest` - Whether to disable rest (TRUE) or not (FALSE)
-

GetAreaNoRest_xpAM

```
int GetAreaNoRest_xpAM(object oArea);
```

Get the possibility of Rest in oArea

Parameters

- `oArea` - The area to get the rest restriction for

Return Value

TRUE if rest is disabled, FALSE otherwise.

SetAreaPVPMode_xpAM

```
void SetAreaPVPMode_xpAM(object oArea, int iPvPMode);
```

Set the PVP Mode of oArea

Parameters

- `oArea` - The area to set the PVP mode for
 - `iPvPMode` - The PVP mode to set ([PVP_SETTING *](#) constant)
-

GetAreaPVPMode_xpAM

```
int GetAreaPVPMode_xpAM(object oArea);
```

Get the PVP Mode of oArea

Parameters

- `oArea` - The area to get the PVP mode for

Return Value

The PVP mode ([PVP_SETTING *](#) constant).

SetAreaModSpotCheck_xpAM

```
void SetAreaModSpotCheck_xpAM(object oArea, int iModSpot);
```

Set the Spot Modifier for this Area

Parameters

- `oArea` - The area to set the spot modifier for
 - `iModSpot` - The spot modifier to set
-

GetAreaModSpotCheck_xpAM

```
int GetAreaModSpotCheck_xpAM(object oArea);
```

Get the Spot Modificator for this Area

Parameters

- `oArea` - The area to get the spot modifier for

Return Value

The spot modifier.

SetAreaModListenCheck_xpAM

```
void SetAreaModListenCheck_xpAM(object oArea, int iModListen);
```

Set the Listen Modificator for this Area

Parameters

- `oArea` - The area to set the listen modifier for
 - `iModListen` - The listen modifier to set
-

GetAreaModListenCheck_xpAM

```
int GetAreaModListenCheck_xpAM(object oArea);
```

Get the Listen Modificator for this Area

Parameters

- `oArea` - The area to get the listen modifier for

Return Value

The listen modifier.

SetAreaSkyRing_xpAM

```
void SetAreaSkyRing_xpAM(object oArea, string sSkyRing, int iVal);
```

Set the SkyRing for oArea

Parameters

- oArea - The area to set the sky ring for
 - sSkyRing - The sky ring to set ("N", "S", "E" or "W")
 - iVal - The desired Skyring.2DA id
-

GetAreaSkyRing_xpAM

```
int GetAreaSkyRing_xpAM(object oArea, string sSkyRing);
```

Get the SkyRing for oArea

Parameters

- oArea - The area to get the sky ring for
- sSkyRing - The sky ring to get ("N", "S", "E" or "W")

Return Value

The Skyring.2DA id.

SetAreaHasDirectionalLight_xpAM

```
void SetAreaHasDirectionalLight_xpAM(object oArea, int bHasDirLight);
```

Enable or Disable the DirectionalLight for oArea

Parameters

- oArea - The area to set the directional light for
 - bHasDirLight - Whether to enable (TRUE) or disable (FALSE) the directional light
-

GetAreaHasDirectionalLight_xpAM

```
int GetAreaHasDirectionalLight_xpAM(object oArea);
```

Get the status of the DirectionalLight of oArea

Parameters

- oArea - The area to get the directional light status for

Return Value

TRUE if the directional light is enabled, FALSE otherwise.

SetAreaIsDirLightCastShadow_xpAM

```
void SetAreaIsDirLightCastShadow_xpAM(object oArea, int bDirLShadow);
```

Choose if the area DirectionalLight cast Shadow or not

Parameters

- `oArea` - The area to set the directional light shadow for
 - `bDirLShadow` - Whether the directional light should cast shadows (TRUE) or not (FALSE)
-

GetAreaIsDirLightCastShadow_xpAM

```
int GetAreaIsDirLightCastShadow_xpAM(object oArea);
```

Get if the area DirectionalLight cast shadows or not

Parameters

- `oArea` - The area to get the directional light shadow status for

Return Value

TRUE if the directional light casts shadows, FALSE otherwise.

Day/Night Cycle

SetAreaHasDayNightCycle_xpAM

```
void SetAreaHasDayNightCycle_xpAM(object oArea, int bHasDNC);
```

Enable or Disable the Day/Night Cycle for oArea

Parameters

- `oArea` - The area to set the day/night cycle for
 - `bHasDNC` - Whether to enable (TRUE) or disable (FALSE) the day/night cycle
-

GetAreaHasDayNightCycle_xpAM

```
int GetAreaHasDayNightCycle_xpAM(object oArea);
```

Get the activation status of Day/Night Cycle for oArea

Parameters

- `oArea` - The area to get the day/night cycle status for

Return Value

TRUE if the day/night cycle is enabled, FALSE otherwise.

SetAreaIsAlwaysNight_xpAM

```
void SetAreaIsAlwaysNight_xpAM(object oArea, int bNight);
```

If Day/Night Cycle is set to false, choose if oArea is always in Night not

Parameters

- `oArea` - The area to set the night status for
 - `bNight` - Whether the area should be in night (TRUE) or not (FALSE)
-

GetAreaIsAlwaysNight_xpAM

```
int GetAreaIsAlwaysNight_xpAM(object oArea);
```

If Day/Night Cycle is set to false, get if oArea is always in Night not

Parameters

- `oArea` - The area to get the night status for

Return Value

TRUE if the area is always in night, FALSE otherwise.

SetAreaUseDayNightFog_xpAM

```
void SetAreaUseDayNightFog_xpAM(object oArea, int bUsedNF);
```

If set to TRUE then the fog color will update as time passes.

Parameters

- `oArea` - The area to set the day/night fog for
 - `bUsedNF` - Whether to enable (TRUE) or disable (FALSE) the day/night fog
-

GetAreaUseDayNightFog_xpAM

```
int GetAreaUseDayNightFog_xpAM(object oArea);
```

Get if fog color update with cycle or not

Parameters

- `oArea` - The area to get the day/night fog status for

Return Value

TRUE if the day/night fog is enabled, FALSE otherwise.

SetAreaDNCCoronaIntensity_xpAM

```
void SetAreaDNCCoronaIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Sun (or moon) Corona Intensity during iDNC

Parameters

- `oArea` - The area to set the corona intensity for
 - `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
 - `fIntensity` - The intensity to set
-

GetAreaDNCCoronaIntensity_xpAM

```
float GetAreaDNCCoronaIntensity_xpAM(object oArea, int iDNC);
```

Get the Sun (or moon) Corona Intensity during this iDNC

Parameters

- `oArea` - The area to get the corona intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The corona intensity.

SetAreaDNCSshadowIntensity_xpAM

```
void SetAreaDNCSshadowIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Shadow Intensity during iDNC

Parameters

- `oArea` - The area to set the shadow intensity for
 - `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
 - `fIntensity` - The intensity to set
-

GetAreaDNCSShadowIntensity_xpAM

```
float GetAreaDNCSShadowIntensity_xpAM(object oArea, int iDNC);
```

Get the Shadow Intensity during iDNC

Parameters

- `oArea` - The area to get the shadow intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The shadow intensity.

SetAreaDNCSkyRingColorInterpretation_xpAM

```
void SetAreaDNCSkyRingColorInterpretation_xpAM(object oArea, int iDNC, float fLerp);
```

Set the "speed" of transition between Horizon and Zenith Color

Parameters

- `oArea` - The area to set the sky ring color interpretation for
 - `iDNC` - The time of day to set the interpretation for ([DAYNIGHT *](#) constant)
 - `fLerp` - The "speed" of transition to set
-

GetAreaDNCSkyRingColorInterpretation_xpAM

```
float GetAreaDNCSkyRingColorInterpretation_xpAM(object oArea, int iDNC);
```

Get the "speed" of transition between Horizon and Zenith Color

Parameters

- `oArea` - The area to get the sky ring color interpretation for
- `iDNC` - The time of day to get the interpretation for ([DAYNIGHT *](#) constant)

Return Value

The "speed" of transition.

SetAreaDNCSkyDomeModel_xpAM

```
void SetAreaDNCSkyDomeModel_xpAM(object oArea, int iDNC, string sSkyDome);
```

Set the SkyDomeModel of oArea during iDNC

Parameters

- oArea - The area to set the sky dome model for
 - iDNC - The time of day to set the model for ([DAYNIGHT *](#) constant)
 - sSkyDome - The sky dome model to set
-

GetAreaDNCSkyDomeModel_xpAM

```
string GetAreaDNCSkyDomeModel_xpAM(object oArea, int iDNC);
```

Set the SkyDomeModel of oArea during iDNC

Parameters

- oArea - The area to get the sky dome model for
- iDNC - The time of day to get the model for ([DAYNIGHT *](#) constant)

Return Value

The sky dome model.

Bloom

SetAreaDNCBloomBLRadius_xpAM

```
void SetAreaDNCBloomBLRadius_xpAM(object oArea, int iDNC, float fRadius);
```

Set the Bloom Blur Radius

Parameters

- oArea - The area to set the bloom blur radius for
 - iDNC - The time of day to set the radius for ([DAYNIGHT *](#) constant)
 - fRadius - The bloom blur radius to set
-

GetAreaDNCBloomBLRadius_xpAM

```
float GetAreaDNCBloomBLRadius_xpAM(object oArea, int iDNC);
```


Get the Bloom Blur Radius

Parameters

- `oArea` - The area to get the bloom blur radius for
- `iDNC` - The time of day to get the radius for ([DAYNIGHT *](#) constant)

Return Value

The bloom blur radius.

SetAreaDNCBloomGLIntensity_xpAM

```
void SetAreaDNCBloomGLIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Bloom Glow Intensity

Parameters

- `oArea` - The area to set the bloom glow intensity for
 - `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
 - `fIntensity` - The bloom glow intensity to set
-

GetAreaDNCBloomGLIntensity_xpAM

```
float GetAreaDNCBloomGLIntensity_xpAM(object oArea, int iDNC);
```

Get the Bloom Glow Intensity

Parameters

- `oArea` - The area to get the bloom glow intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The bloom glow intensity.

SetAreaDNCBloomHLIntensity_xpAM

```
void SetAreaDNCBloomHLIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Bloom Highlight Intensity

Parameters

- `oArea` - The area to set the bloom highlight intensity for
- `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)

- `fIntensity` - The bloom highlight intensity to set
-

GetAreaDNCBloomHLIntensity_xpAM

```
float GetAreaDNCBloomHLIntensity_xpAM(object oArea, int iDNC);
```

Get the Bloom Highlight Intensity

Parameters

- `oArea` - The area to get the bloom highlight intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT](#) * constant)

Return Value

The bloom highlight intensity.

SetAreaDNCBloomHLThreshold_xpAM

```
void SetAreaDNCBloomHLThreshold_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Bloom Highlight Threshold

Parameters

- `oArea` - The area to set the bloom highlight threshold for
 - `iDNC` - The time of day to set the threshold for ([DAYNIGHT](#) * constant)
 - `fIntensity` - The bloom highlight threshold to set
-

GetAreaDNCBloomHLThreshold_xpAM

```
float GetAreaDNCBloomHLThreshold_xpAM(object oArea, int iDNC);
```

Get the Bloom Highlight Threshold

Parameters

- `oArea` - The area to get the bloom highlight threshold for
- `iDNC` - The time of day to get the threshold for ([DAYNIGHT](#) * constant)

Return Value

The bloom highlight threshold.

SetAreaDNCBloomSCIntensity_xpAM

```
void SetAreaDNCBloomSCIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Bloom Scene Intensity

Parameters

- `oArea` - The area to set the bloom scene intensity for
 - `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
 - `fIntensity` - The bloom scene intensity to set
-

GetAreaDNCBloomSCIntensity_xpAM

```
float GetAreaDNCBloomSCIntensity_xpAM(object oArea, int iDNC);
```

Get the Bloom Scene Intensity

Parameters

- `oArea` - The area to get the bloom scene intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The bloom scene intensity.

HDR

SetAreaDNCHDRAvg_xpAM

```
void SetAreaDNCHDRAvg_xpAM(object oArea, int iDNC, float fHDR);
```

Set the HDR Average Value

Parameters

- `oArea` - The area to set the HDR average value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fHDR` - The HDR average value to set
-

GetAreaDNCHDRAvg_xpAM

```
float GetAreaDNCHDRAvg_xpAM(object oArea, int iDNC);
```

Get the HDR Average Value

Parameters

- `oArea` - The area to get the HDR average value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The HDR average value.

SetAreaDNCHDRMax_xpAM

```
void SetAreaDNCHDRMax_xpAM(object oArea, int iDNC, float fHDR);
```

Set the HDR Maximum Value

Parameters

- `oArea` - The area to set the HDR maximum value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fHDR` - The HDR maximum value to set
-

GetAreaDNCHDRMax_xpAM

```
float GetAreaDNCHDRMax_xpAM(object oArea, int iDNC);
```

Get the HDR Maximum Value

Parameters

- `oArea` - The area to get the HDR maximum value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The HDR maximum value.

SetAreaDNCHDRExposure_xpAM

```
void SetAreaDNCHDRExposure_xpAM(object oArea, int iDNC, float fHDR);
```

Set the HDR Exposure Value

Parameters

- `oArea` - The area to set the HDR exposure value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fHDR` - The HDR exposure value to set
-

GetAreaDNCHDRExposure_xpAM

```
float GetAreaDNCHDRExposure_xpAM(object oArea, int iDNC);
```

Get the HDR Exposure Value

Parameters

- `oArea` - The area to get the HDR exposure value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The HDR exposure value.

SetAreaDNCHDRHighlight_xpAM

```
void SetAreaDNCHDRHighlight_xpAM(object oArea, int iDNC, float fHDR);
```

Set the HDR Highlight Value

Parameters

- `oArea` - The area to set the HDR highlight value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fHDR` - The HDR highlight value to set
-

GetAreaDNCHDRHighlight_xpAM

```
float GetAreaDNCHDRHighlight_xpAM(object oArea, int iDNC);
```

Get the HDR Highlight Value

Parameters

- `oArea` - The area to get the HDR highlight value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The HDR highlight value.

Cloud

SetAreaDNCCloudCover_xpAM

```
void SetAreaDNCCloudCover_xpAM(object oArea, int iDNC, float fCover);
```

Set the Cloud Cover

Parameters

- `oArea` - The area to set the cloud cover for
 - `iDNC` - The time of day to set the cover for ([DAYNIGHT *](#) constant)
 - `fCover` - The cloud cover to set (value between 0 and 1.5)
-

GetAreaDNCCloudCover_xpAM

```
float GetAreaDNCCloudCover_xpAM(object oArea, int iDNC);
```

Get the Cloud Cover

Parameters

- `oArea` - The area to get the cloud cover for
- `iDNC` - The time of day to get the cover for ([DAYNIGHT *](#) constant)

Return Value

The cloud cover (value between 0 and 1.5).

SetAreaDNCCloudMoveX_xpAM

```
void SetAreaDNCCloudMoveX_xpAM(object oArea, int iDNC, float fMove);
```

Set the Cloud Movement in the East-West direction

Parameters

- `oArea` - The area to set the cloud movement for
- `iDNC` - The time of day to set the movement for ([DAYNIGHT *](#) constant)
- `fMove` - The cloud movement in the East-West direction to set

Note

A positive values causing movement toward the west. Unless the weather is extreme, you will likely want values of 0.1 or less.

GetAreaDNCCloudMoveX_xpAM

```
float GetAreaDNCCloudMoveX_xpAM(object oArea, int iDNC);
```

Get the Cloud Movement in the East-West direction

Parameters

- `oArea` - The area to get the cloud movement for
- `iDNC` - The time of day to get the movement for ([DAYNIGHT](#) * constant)

Return Value

The cloud movement in the East-West direction. A positive value represent a movement toward the west.

SetAreaDNCCloudMoveY_xpAM

```
void SetAreaDNCCloudMoveY_xpAM(object oArea, int iDNC, float fMove);
```

Set the Cloud Movement in the North-South direction

Parameters

- `oArea` - The area to set the cloud movement for
- `iDNC` - The time of day to set the movement for ([DAYNIGHT](#) * constant)
- `fMove` - The cloud movement in the North-South direction to set

Note

A positive values causing movement toward the South. Unless the weather is extreme, you will likely want values of 0.1 or less.

GetAreaDNCCloudMoveY_xpAM

```
float GetAreaDNCCloudMoveY_xpAM(object oArea, int iDNC);
```

Get the Cloud Movement in the North-South direction

Parameters

- `oArea` - The area to get the cloud movement for
- `iDNC` - The time of day to get the movement for ([DAYNIGHT](#) * constant)

Return Value

The cloud movement in the North-South direction. A positive value represent a movement toward the South.

Desaturate

SetAreaDNCDesaturateColor_xpAM

```
void SetAreaDNCDesaturateColor_xpAM(object oArea, int iDNC, int bDesaturate);
```

Enable or Disable the Desaturation effect

Parameters

- `oArea` - The area to set the desaturation for
 - `iDNC` - The time of day to set the effect for ([DAYNIGHT *](#) constant)
 - `bDesaturate` - Whether to enable (TRUE) or disable (FALSE) the desaturation effect
-

GetAreaDNCDesaturateColor_xpAM

```
int GetAreaDNCDesaturateColor_xpAM(object oArea, int iDNC);
```

Get the status of desaturation effect

Parameters

- `oArea` - The area to get the desaturation status for
- `iDNC` - The time of day to get the status for ([DAYNIGHT *](#) constant)

Return Value

TRUE if desaturation is enabled, FALSE otherwise.

SetAreaDNCDesaturateFactor_xpAM

```
void SetAreaDNCDesaturateFactor_xpAM(object oArea, int iDNC, float fFactor);
```

Set the desaturation factor

Parameters

- `oArea` - The area to set the desaturation factor for
- `iDNC` - The time of day to set the factor for ([DAYNIGHT *](#) constant)
- `fFactor` - The desaturation factor to set (value from 0 to 1)

Note

If `DesaturateColor` is TRUE, then this value controls the amount of leeching of the color.

GetAreaDNCDesaturateFactor_xpAM

```
float GetAreaDNCDesaturateFactor_xpAM(object oArea, int iDNC);
```

Get the desaturation factor

Parameters

- `oArea` - The area to get the desaturation factor for
- `iDNC` - The time of day to get the factor for ([DAYNIGHT *](#) constant)

Return Value

The desaturation factor (value from 0 to 1).

GroundLight

SetAreaDNCGroundLightIntensity_xpAM

```
void SetAreaDNCGroundLightIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the GroundLight Intensity

Parameters

- `oArea` - The area to set the ground light intensity for
- `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
- `fIntensity` - The ground light intensity to set

GetAreaDNCGroundLightIntensity_xpAM

```
float GetAreaDNCGroundLightIntensity_xpAM(object oArea, int iDNC);
```

Get the GroundLight Intensity

Parameters

- `oArea` - The area to get the ground light intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The ground light intensity.

SetAreaDNCGroundLightColor_xpAM

```
void SetAreaDNCGroundLightColor_xpAM(object oArea, int iDNC, string sColorType, string sRGB);
```

Set the GroundLight Color

Parameters

- `oArea` - The area to set the ground light color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sColorType` - The type of color to set ("Ambient", "Diffuse", or "Specular")
 - `sRGB` - The RGB value of the color to set (on #RRGGBB format)
-

GetAreaDNCGroundLightColor_xpAM

```
string GetAreaDNCGroundLightColor_xpAM(object oArea, int iDNC, string sColorType);
```

Get the GroundLight Color

Parameters

- `oArea` - The area to get the ground light color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)
- `sColorType` - The type of color to get ("Ambient", "Diffuse", or "Specular")

Return Value

The RGB value of the ground light color (on #RRGGBB format).

SkyLight

SetAreaDNCSkylightIntensity_xpAM

```
void SetAreaDNCSkylightIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the SkyLight Intensity

Parameters

- `oArea` - The area to set the skylight intensity for
 - `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)
 - `fIntensity` - The skylight intensity to set
-

GetAreaDNCSkylightIntensity_xpAM

```
float GetAreaDNCSkylightIntensity_xpAM(object oArea, int iDNC);
```

Get the SkyLight Intensity

Parameters

- `oArea` - The area to get the skylight intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The skylight intensity.

SetAreaDNCSkylightColor_xpAM

```
void SetAreaDNCSkylightColor_xpAM(object oArea, int iDNC, string sColorType, string sRGB);
```

Set the SkyLight Color

Parameters

- `oArea` - The area to set the skylight color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sColorType` - The type of color to set ("Ambient", "Diffuse", or "Specular")
 - `sRGB` - The RGB value of the color to set (in #RRGGGBB format)
-

GetAreaDNCSkylightColor_xpAM

```
string GetAreaDNCSkylightColor_xpAM(object oArea, int iDNC, string sColorType);
```

Get the SkyLight Color

Parameters

- `oArea` - The area to get the skylight color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)
- `sColorType` - The type of color to get ("Ambient", "Diffuse", or "Specular")

Return Value

The RGB value of the skylight color (in #RRGGGBB format).

SunMoon

SetAreaDNCSunMoonIntensity_xpAM

```
void SetAreaDNCSunMoonIntensity_xpAM(object oArea, int iDNC, float fIntensity);
```

Set the Sun/Moon Intensity

Parameters

- `oArea` - The area to set the sun/moon intensity for
- `iDNC` - The time of day to set the intensity for ([DAYNIGHT *](#) constant)

- `fIntensity` - The sun/moon intensity to set
-

GetAreaDNCSunMoonIntensity_xpAM

```
float GetAreaDNCSunMoonIntensity_xpAM(object oArea, int iDNC);
```

Get the Sun/Moon Intensity

Parameters

- `oArea` - The area to get the sun/moon intensity for
- `iDNC` - The time of day to get the intensity for ([DAYNIGHT *](#) constant)

Return Value

The sun/moon intensity.

SetAreaDNCSunMoonColor_xpAM

```
void SetAreaDNCSunMoonColor_xpAM(object oArea, int iDNC, string sColorType, string sRGB);
```

Set the Sun/Moon Color

Parameters

- `oArea` - The area to set the sun/moon color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sColorType` - The type of color to set ("Ambient", "Diffuse", or "Specular")
 - `sRGB` - The RGB value of the color to set (in #RRGGGBB format)
-

GetAreaDNCSunMoonColor_xpAM

```
string GetAreaDNCSunMoonColor_xpAM(object oArea, int iDNC, string sColorType);
```

Get the Sun/Moon Color

Parameters

- `oArea` - The area to get the sun/moon color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)
- `sColorType` - The type of color to get ("Ambient", "Diffuse", or "Specular")

Return Value

The RGB value of the sun/moon color (in #RRGGGBB format).

SetAreaDNCSunMoonDirX_xpAM

```
void SetAreaDNCSunMoonDirX_xpAM(object oArea, int iDNC, float fDir);
```

Set the Sun/Moon Direction in the X axis

Parameters

- `oArea` - The area to set the sun/moon direction for
 - `iDNC` - The time of day to set the direction for ([DAYNIGHT *](#) constant)
 - `fDir` - The sun/moon direction in the X axis to set
-

GetAreaDNCSunMoonDirX_xpAM

```
float GetAreaDNCSunMoonDirX_xpAM(object oArea, int iDNC);
```

Get the Sun/Moon Direction in the X axis

Parameters

- `oArea` - The area to get the sun/moon direction for
- `iDNC` - The time of day to get the direction for ([DAYNIGHT *](#) constant)

Return Value

The sun/moon direction in the X axis.

SetAreaDNCSunMoonDirY_xpAM

```
void SetAreaDNCSunMoonDirY_xpAM(object oArea, int iDNC, float fDir);
```

Set the Sun/Moon Direction in the Y axis

Parameters

- `oArea` - The area to set the sun/moon direction for
 - `iDNC` - The time of day to set the direction for ([DAYNIGHT *](#) constant)
 - `fDir` - The sun/moon direction in the Y axis to set
-

GetAreaDNCSunMoonDirY_xpAM

```
float GetAreaDNCSunMoonDirY_xpAM(object oArea, int iDNC);
```

Get the Sun/Moon Direction in the Y axis

Parameters

- `oArea` - The area to get the sun/moon direction for
- `iDNC` - The time of day to get the direction for ([DAYNIGHT *](#) constant)

Return Value

The sun/moon direction in the Y axis.

SetAreaDNCSunMoonDirZ_xpAM

```
void SetAreaDNCSunMoonDirZ_xpAM(object oArea, int iDNC, float fDir);
```

Set the Sun/Moon Direction in the Z axis

Parameters

- `oArea` - The area to set the sun/moon direction for
 - `iDNC` - The time of day to set the direction for ([DAYNIGHT *](#) constant)
 - `fDir` - The sun/moon direction in the Z axis to set
-

GetAreaDNCSunMoonDirZ_xpAM

```
float GetAreaDNCSunMoonDirZ_xpAM(object oArea, int iDNC);
```

Get the Sun/Moon Direction in the Z axis

Parameters

- `oArea` - The area to get the sun/moon direction for
- `iDNC` - The time of day to get the direction for ([DAYNIGHT *](#) constant)

Return Value

The sun/moon direction in the Z axis.

Fog

SetAreaDNCFogStart_xpAM

```
void SetAreaDNCFogStart_xpAM(object oArea, int iDNC, float fStart);
```

Set the Fog Start value

Parameters

- `oArea` - The area to set the fog start value for

- `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fStart` - The fog start value to set
-

GetAreaDNCFogStart_xpAM

```
float GetAreaDNCFogStart_xpAM(object oArea, int iDNC);
```

Get the Fog Start value

Parameters

- `oArea` - The area to get the fog start value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The fog start value.

SetAreaDNCFogEnd_xpAM

```
void SetAreaDNCFogEnd_xpAM(object oArea, int iDNC, float fEnd);
```

Set the Fog End value

Parameters

- `oArea` - The area to set the fog end value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fEnd` - The fog end value to set
-

GetAreaDNCFogEnd_xpAM

```
float GetAreaDNCFogEnd_xpAM(object oArea, int iDNC);
```

Get the Fog End value

Parameters

- `oArea` - The area to get the fog end value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The fog end value.

SetAreaDNCFogFarClip_xpAM

```
void SetAreaDNCFogFarClip_xpAM(object oArea, int iDNC, float fFarClip);
```

Set the Fog Far Clip value

Parameters

- `oArea` - The area to set the fog far clip value for
 - `iDNC` - The time of day to set the value for ([DAYNIGHT *](#) constant)
 - `fFarClip` - The fog far clip value to set
-

GetAreaDNCFogFarClip_xpAM

```
float GetAreaDNCFogFarClip_xpAM(object oArea, int iDNC);
```

Get the Fog Far Clip value

Parameters

- `oArea` - The area to get the fog far clip value for
- `iDNC` - The time of day to get the value for ([DAYNIGHT *](#) constant)

Return Value

The fog far clip value.

SetAreaDNCFogColor_xpAM

```
void SetAreaDNCFogColor_xpAM(object oArea, int iDNC, string sRGB);
```

Set the Fog Color

Parameters

- `oArea` - The area to set the fog color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sRGB` - The RGB value of the color to set (in #RRGGGBB format)
-

GetAreaDNCFogColor_xpAM

```
string GetAreaDNCFogColor_xpAM(object oArea, int iDNC);
```

Get the Fog Color

Parameters

- `oArea` - The area to get the fog color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)

Return Value

The RGB value of the fog color (in #RRGGGBB format).

Other Colors

SetAreaDNCSkyHorizonColor_xpAM

```
void SetAreaDNCSkyHorizonColor_xpAM(object oArea, int iDNC, string sRGB);
```

Set the SkyHorizon Color

Parameters

- `oArea` - The area to set the SkyHorizon color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sRGB` - The RGB value of the color to set (in #RRGGGBB format)
-

GetAreaDNCSkyHorizonColor_xpAM

```
string GetAreaDNCSkyHorizonColor_xpAM(object oArea, int iDNC);
```

Get the SkyHorizon Color

Parameters

- `oArea` - The area to get the SkyHorizon color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)

Return Value

The RGB value of the SkyHorizon color (in #RRGGGBB format).

SetAreaDNCSkyZenithColor_xpAM

```
void SetAreaDNCSkyZenithColor_xpAM(object oArea, int iDNC, string sRGB);
```

Set the SkyZenith Color

Parameters

- `oArea` - The area to set the SkyZenith color for
 - `iDNC` - The time of day to set the color for ([DAYNIGHT *](#) constant)
 - `sRGB` - The RGB value of the color to set (in #RRGGGBB format)
-

GetAreaDNCSkyZenithColor_xpAM

```
string GetAreaDNCSkyZenithColor_xpAM(object oArea, int iDNC);
```

Get the SkyZenith Color

Parameters

- `oArea` - The area to get the SkyZenith color for
- `iDNC` - The time of day to get the color for ([DAYNIGHT *](#) constant)

Return Value

The RGB value of the SkyZenith color (in #RRGGGBB format).

Constants Area

PVP Setting (PVP_SETTING_*)

Name	Value	Description
PVP_SETTING_NO	0	
PVP_SETTING_GROUP	1	
PVP_SETTING_TOTAL	2	
PVP_SETTING_SERVER	3	

Area Flag (AREA_FLAG_*)

Name	Value	Description
AREA_FLAG_INTERIOR	0	
AREA_FLAG_SUBTERRAN	2	
AREA_FLAG_NATURAL	4	

DayNights (DAYNIGHT_*)

Name	Value	Description
DAYNIGHT_SUNRISE	0	
DAYNIGHT_DAYTIME	1	
DAYNIGHT_SUNSET	2	
DAYNIGHT_MOONRISE	3	
DAYNIGHT_NIGHTTIME	4	
DAYNIGHT_MOONSET	5	
DAYNIGHT_TRANSITION	6	
DAYNIGHT_DEFAULT	7	

xp_aspectManager_item.nss

- [SetBaseItemType_xpAM](#)
- [SetItemAppearanceVariation_xpAM](#)
- [GetItemAppearanceVariation_xpAM](#)
- [SetItemAppearanceCategory_xpAM](#)
- [GetItemAppearanceCategory_xpAM](#)
- [SetItemDescription_xpAM](#)
- [GetItemDescription_xpAM](#)
- [SetItemColor_xpAM](#)
- [GetItemColor_xpAM](#)
- **[WeaponSpecific](#)**
 - [SetItemWeaponFX_xpAM](#)
 - [GetItemWeaponFX_xpAM](#)
 - [SetItemWeaponPart_xpAM](#)
 - [GetItemWeaponPart_xpAM](#)
- **[ModelPiece](#)**
 - [SetItemModelPieceVariation_xpAM](#)
 - [GetItemModelPieceVariation_xpAM](#)
 - [SetItemModelPieceColor_xpAM](#)
 - [GetItemModelPieceColor_xpAM](#)
 - [SetItemModelPieceIsUVScroll_xpAM](#)
 - [GetItemModelPieceIsUVScroll_xpAM](#)
 - [SetItemModelPieceUScroll_xpAM](#)
 - [GetItemModelPieceUScroll_xpAM](#)
 - [SetItemModelPieceVScroll_xpAM](#)
 - [GetItemModelPieceVScroll_xpAM](#)
- **[ModelPart](#)**
 - [SetItemModelPartVariation_xpAM](#)
 - [GetItemModelPartVariation_xpAM](#)
 - [SetItemModelPartCategory_xpAM](#)
 - [GetItemModelPartCategory_xpAM](#)
 - [SetItemModelPartColor_xpAM](#)
 - [GetItemModelPartColor_xpAM](#)

SetBaseItemType_xpAM

```
void SetBaseItemType_xpAM(  
    object oItem,  
    int iItemType  
);
```

Set the base item type (BASE_ITEM_* or Baseitems.2da ID) of oItem.

Parameters

- `oItem` - The item to set the base type for
 - `iItemType` - The base item type to set
-

SetItemAppearanceVariation_xpAM

```
void SetItemAppearanceVariation_xpAM(  
    object oItem,  
    int iVariation  
);
```

Set the Visual Variation of oItem

Parameters

- `oItem` - The item to set the visual variation for
 - `iVariation` - The visual variation to set
-

GetItemAppearanceVariation_xpAM

```
int GetItemAppearanceVariation_xpAM(  
    object oItem  
);
```

Get the Visual Variation of oItem

Parameters

- `oItem` - The item to get the visual variation for

Return Value

The visual variation of oItem

SetItemAppearanceCategory_xpAM

```
void SetItemAppearanceCategory_xpAM(  
    object oItem,  
    int iCategory  
);
```

Set the Visual Category of oItem (armorvisualdata.2DA ID)

Parameters

- `oItem` - The item to set the visual category for
 - `iCategory` - The visual category to set
-

GetItemAppearanceCategory_xpAM

```
int GetItemAppearanceCategory_xpAM(  
    object oItem  
);
```

Get the Visual Category of oItem (armorvisualdata.2DA ID)

Parameters

- `oItem` - The item to get the visual category for

Return Value

The visual category of oItem

SetItemDescription_xpAM

```
void SetItemDescription_xpAM(  
    object oItem,  
    string sDescription,  
    int bIdentified = TRUE  
);
```

Set the Description of oItem.

Parameters

- `oItem` - The item to set the description for
 - `sDescription` - The description to set
 - `bIdentified` - If TRUE, set the identified description. If FALSE, set the non-identified description.
-

GetItemDescription_xpAM

```
string GetItemDescription_xpAM(  
    object oItem,  
    int bIdentified = TRUE  
);
```

Get the Description of oItem.

Parameters

- `oItem` - The item to get the description for
- `bIdentified` - If TRUE, get the identified description. If FALSE, get the non-identified description.

Return Value

The description of oItem

SetItemColor_xpAM

```
void SetItemColor_xpAM(  
    object oItem,  
    int iColor,  
    string sRGB  
);
```

Set the oItem Color.

Parameters

- `oItem` - The item to set the color for
- `iColor` - 1, 2 or 3
- `sRGB` - A RGB string value, on the "#FFFFFF" (or #FFFFFFF) format

GetItemColor_xpAM

```
string GetItemColor_xpAM(  
    object oItem,  
    int iColor  
);
```

Get the oItem Color.

Parameters

- `oItem` - The item to get the color for
- `iColor` - 1, 2 or 3

Return Value

The color of oItem in RGB string format

WeaponSpecific

SetItemWeaponFX_xpAM

```
void SetItemWeaponFX_xpAM(  
    object oItem,  
    int iwpnFX  
);
```

Set the "DamageEffect" of the Weapon (ITEM_VISUAL_* / iprp_visualfx.2da)

Parameters

- `oItem` - The weapon to set the damage effect for
 - `iWpnFX` - The damage effect to set
-

GetItemWeaponFX_xpAM

```
int GetItemWeaponFX_xpAM(  
    object oItem  
);
```

Get the "DamageEffect" of the Weapon (ITEM_VISUAL_* / iprp_visualfx.2da)

Parameters

- `oItem` - The weapon to get the damage effect for

Return Value

The damage effect of the weapon

SetItemWeaponPart_xpAM

```
void SetItemWeaponPart_xpAM(  
    object oItem,  
    int iPart,  
    int iVariation  
);
```

Set the iPart (WEAPON_MODEL_PART_*) of oItem to Variation

Parameters

- `oItem` - The weapon to set the part variation for
 - `iPart` - The part to set the variation for (WEAPON_MODEL_PART_* constant)
 - `iVariation` - The variation to set
-

GetItemWeaponPart_xpAM

```
int GetItemWeaponPart_xpAM(  
    object oItem,  
    int iPart  
);
```

Get the Variation of the iPart (WEAPON_MODEL_PART_*) of oItem

Parameters

- `oItem` - The weapon to get the part variation for

- `iPart` - The part to get the variation for (WEAPON_MODEL_PART_* constant)

Return Value

The variation of the part

ModelPiece

SetItemModelPieceVariation_xpAM

```
void SetItemModelPieceVariation_xpAM(  
    object oItem,  
    int iModelPiece,  
    int iVariation  
);
```

Set the ModelPiece Variation for oItem.

Parameters

- `oItem` - The item to set the model piece variation for
 - `iModelPiece` - The model piece to set the variation for (ARMOR_MODEL_PIECE_* constant)
 - `iVariation` - The variation to set
-

GetItemModelPieceVariation_xpAM

```
int GetItemModelPieceVariation_xpAM(  
    object oItem,  
    int iModelPiece  
);
```

Get the ModelPiece Variation for oItem.

Parameters

- `oItem` - The item to get the model piece variation for
- `iModelPiece` - The model piece to get the variation for (ARMOR_MODEL_PIECE_* constant)

Return Value

The variation of the model piece

SetItemModelPieceColor_xpAM

```
void SetItemModelPieceColor_xpAM(  
    object oItem,  
    int iModelPiece,  
    int iColor,  
    string sRGB  
);
```

Set the Color of the iModelPiece of oItem

Parameters

- `oItem` - The item to set the model piece color for
 - `iModelPiece` - The model piece to set the color for (ARMOR_MODEL_PIECE_* constant)
 - `iColor` - 1, 2 or 3
 - `sRGB` - A RGB string value, on the "#FFFFFF" (or #FFFFFFF) format
-

GetItemModelPieceColor_xpAM

```
string GetItemModelPieceColor_xpAM(  
    object oItem,  
    int iModelPiece,  
    int iColor  
);
```

Get the Color of the iModelPiece of oItem

Parameters

- `oItem` - The item to get the model piece color for
- `iModelPiece` - The model piece to get the color for (ARMOR_MODEL_PIECE_* constant)
- `iColor` - 1, 2 or 3

Return Value

The color of the model piece in RGB string format

SetItemModelPieceIsUVScroll_xpAM

```
void SetItemModelPieceIsUVScroll_xpAM(  
    object oItem,  
    int iModelPiece,  
    int bUVScroll  
);
```

Enable or Disable the UVScroll of the iModelPiece of oItem

Parameters

- `oItem` - The item to enable or disable the UVScroll for
 - `iModelPiece` - The model piece to enable or disable the UVScroll for (ARMOR_MODEL_PIECE_* constant)
 - `bUVScroll` - TRUE to enable UVScroll, FALSE to disable UVScroll
-

GetItemModelPieceIsUVScroll_xpAM

```
int GetItemModelPieceIsUVScroll_xpAM(  
    object oItem,  
    int iModelPiece  
);
```

Get the activation status for the UVScroll of the iModelPiece of oItem

Parameters

- `oItem` - The item to get the UVScroll activation status for
- `iModelPiece` - The model piece to get the UVScroll activation status for (ARMOR_MODEL_PIECE_* constant)

Return Value

TRUE if UVScroll is enabled, FALSE if UVScroll is disabled

SetItemModelPieceUScroll_xpAM

```
void SetItemModelPieceUScroll_xpAM(  
    object oItem,  
    int iModelPiece,  
    float fUScroll  
);
```

Set the UScroll speed for the iModelPiece of oItem

Parameters

- `oItem` - The item to set the UScroll speed for
 - `iModelPiece` - The model piece to set the UScroll speed for (ARMOR_MODEL_PIECE_* constant)
 - `fUScroll` - The UScroll speed to set
-

GetItemModelPieceUScroll_xpAM

```
float GetItemModelPieceUScroll_xpAM(  
    object oItem,
```

```
    int iModelPiece  
);
```

Get the UScroll speed for the iModelPiece of oItem

Parameters

- `oItem` - The item to get the UScroll speed for
- `iModelPiece` - The model piece to get the UScroll speed for (ARMOR_MODEL_PIECE_* constant)

Return Value

The UScroll speed of the model piece

SetItemModelPieceVScroll_xpAM

```
void SetItemModelPieceVScroll_xpAM(  
    object oItem,  
    int iModelPiece,  
    float fVScroll  
);
```

Set the VScroll speed for the iModelPiece of oItem

Parameters

- `oItem` - The item to set the VScroll speed for
 - `iModelPiece` - The model piece to set the VScroll speed for (ARMOR_MODEL_PIECE_* constant)
 - `fVScroll` - The VScroll speed to set
-

GetItemModelPieceVScroll_xpAM

```
float GetItemModelPieceVScroll_xpAM(  
    object oItem,  
    int iModelPiece  
);
```

Get the VScroll speed for the iModelPiece of oItem

Parameters

- `oItem` - The item to get the VScroll speed for
- `iModelPiece` - The model piece to get the VScroll speed for (ARMOR_MODEL_PIECE_* constant)

Return Value

The VScroll speed of the model piece

ModelPart

SetItemModelPartVariation_xpAM

```
void SetItemModelPartVariation_xpAM(  
    object oItem,  
    int iModelPart,  
    int iVariation  
);
```

Set the ModelPart Variation for oItem.

Parameters

- `oItem` - The item to set the model part variation for
- `iModelPart` - The model part to set the variation for (ARMOR_MODEL_PART_* constant)
- `iVariation` - The variation to set

GetItemModelPartVariation_xpAM

```
int GetItemModelPartVariation_xpAM(  
    object oItem,  
    int iModelPart  
);
```

Get the ModelPart Variation for oItem.

Parameters

- `oItem` - The item to get the model part variation for
- `iModelPart` - The model part to get the variation for (ARMOR_MODEL_PART_* constant)

Return Value

The variation of the model part

SetItemModelPartCategory_xpAM

```
void SetItemModelPartCategory_xpAM(  
    object oItem,  
    int iModelPart,  
    int iCategory  
);
```

Set the ModelPart Visual Category for oItem (armorvisualdata.2DA ID).

Parameters

- `oItem` - The item to set the model part visual category for
 - `iModelPart` - The model part to set the visual category for (ARMOR_MODEL_PART_* constant)
 - `iCategory` - The visual category to set
-

GetItemModelPartCategory_xpAM

```
int GetItemModelPartCategory_xpAM(  
    object oItem,  
    int iModelPart  
);
```

Get the ModelPart Visual Category for oItem (armorvisualdata.2DA ID).

Parameters

- `oItem` - The item to get the model part visual category for
- `iModelPart` - The model part to get the visual category for (ARMOR_MODEL_PART_* constant)

Return Value

The visual category of the model part

SetItemModelPartColor_xpAM

```
void SetItemModelPartColor_xpAM(  
    object oItem,  
    int iModelPart,  
    int iColor,  
    string sRGB  
);
```

Set the Color of the iModelPart of oItem

Parameters

- `oItem` - The item to set the model part color for
 - `iModelPart` - The model part to set the color for (ARMOR_MODEL_PART_* constant)
 - `iColor` - 1, 2 or 3
 - `sRGB` - A RGB string value, on the "#FFFFFF" (or #FFFFFF) format
-

GetItemModelPartColor_xpAM

```
string GetItemModelPartColor_xpAM(  
    object oItem,  
    int iModelPart,  
    int iColor  
);
```

Get the Color of the iModelPart of oItem

Parameters

- `oItem` - The item to get the model part color for
- `iModelPart` - The model part to get the color for (ARMOR_MODEL_PART_* constant)
- `iColor` - 1, 2 or 3

Return Value

The color of the model part in RGB string format

xp_aspectManager_light.nss

- [SetLightRange_xpAM](#)
- [GetLightRange_xpAM](#)
- [Shadow](#)
 - [SetLightCastShadow_xpAM](#)
 - [GetLightCastShadow_xpAM](#)
 - [SetLightShadowIntensity_xpAM](#)
 - [GetLightShadowIntensity_xpAM](#)
- [Light](#)
 - [SetLightColorIntensity_xpAM](#)
 - [GetLightColorIntensity_xpAM](#)
 - [SetLightColor_xpAM](#)
 - [GetLightColor_xpAM](#)
- [Lerp](#)
 - [SetLightIsLerp_xpAM](#)
 - [GetLightIsLerp_xpAM](#)
 - [SetLightLerpPeriod_xpAM](#)
 - [GetLightLerpPeriod_xpAM](#)
 - [SetLightLerpIntensity_xpAM](#)
 - [GetLightLerpIntensity_xpAM](#)
 - [SetLightLerpColor_xpAM](#)
 - [GetLightLerpColor_xpAM](#)
- [Flicker](#)
 - [SetLightIsFlicker_xpAM](#)
 - [GetLightIsFlicker_xpAM](#)
 - [SetLightFlickerType_xpAM](#)
 - [GetLightFlickerType_xpAM](#)
 - [SetLightFlickerVariance_xpAM](#)
 - [GetLightFlickerVariance_xpAM](#)
 - [SetLightFlickerRate_xpAM](#)
 - [GetLightFlickerRate_xpAM](#)

SetLightRange_xpAM

```
void SetLightRange_xpAM(  
    object oLight,  
    float fRange  
);
```

Set the maximum distance that the light's diffuse color will reach.

Parameters

- oLight - The light object
 - fRange - The maximum distance for the light's diffuse color
-

GetLightRange_xpAM

```
float GetLightRange_xpAM(  
    object oLight  
);
```

Get the maximum distance that the light's diffuse color will reach.

Parameters

- `oLight` - The light object

Return Value

The maximum distance that the light's diffuse color will reach.

Shadow

SetLightCastShadow_xpAM

```
void SetLightCastShadow_xpAM(  
    object oLight,  
    int bCastShadow  
);
```

Set if the light should cast shadows or not.

Parameters

- `oLight` - The light object
 - `bCastShadow` - True if the light should cast shadows, false otherwise
-

GetLightCastShadow_xpAM

```
int GetLightCastShadow_xpAM(  
    object oLight  
);
```

Get if the light casts shadows or not.

Parameters

- `oLight` - The light object

Return Value

True if the light casts shadows, false otherwise.

SetLightShadowIntensity_xpAM

```
void SetLightShadowIntensity_xpAM(  
    object oLight,  
    float fIntensity  
);
```

Set the intensity of shadows cast by the light.

Parameters

- `oLight` - The light object
- `fIntensity` - The shadow intensity

GetLightShadowIntensity_xpAM

```
float GetLightShadowIntensity_xpAM(  
    object oLight  
);
```

Get the intensity of shadows cast by the light.

Parameters

- `oLight` - The light object

Return Value

The intensity of shadows cast by the light.

Light

SetLightColorIntensity_xpAM

```
void SetLightColorIntensity_xpAM(  
    object oLight,  
    float fColorIntensity  
);
```

Set the intensity (brightness) of the light within its range.

Parameters

- `oLight` - The light object
 - `fColorIntensity` - The light color intensity
-

GetLightColorIntensity_xpAM

```
float GetLightColorIntensity_xpAM(  
    object oLight  
);
```

Get the intensity (brightness) of the light within its range.

Parameters

- `oLight` - The light object

Return Value

The intensity (brightness) of the light within its range.

SetLightColor_xpAM

```
void SetLightColor_xpAM(  
    object oLight,  
    string sColorType,  
    string sRGB  
);
```

Set the chosen light color value.

Parameters

- `oLight` - The light object
 - `sColorType` - The [LIGHT_COLOR_TYPE](#) * constant
 - `sRGB` - The RGB string value in the "#FFFFFF" (or #FFFFFFF) format
-

GetLightColor_xpAM

```
string GetLightColor_xpAM(  
    object oLight,  
    string sColorType  
);
```

Get the chosen light color value.

Parameters

- `oLight` - The light object
- `sColorType` - The [LIGHT_COLOR_TYPE](#) * constant

Return Value

The RGB string value of the chosen light color.

Lerp

SetLightIsLerp_xpAM

```
void SetLightIsLerp_xpAM(  
    object oLight,  
    int bLerp  
);
```

Enable or disable lerp for this light.

Parameters

- `oLight` - The light object
 - `bLerp` - True to enable lerp, false to disable
-

GetLightIsLerp_xpAM

```
int GetLightIsLerp_xpAM(  
    object oLight  
);
```

Get the status of lerp for this light.

Parameters

- `oLight` - The light object

Return Value

True if lerp is enabled for the light, false otherwise.

SetLightLerpPeriod_xpAM

```
void SetLightLerpPeriod_xpAM(  
    object oLight,  
    float fLerpPeriod  
);
```

Set the time in seconds for the light to cycle between the color and lerp target color.

Parameters

- `oLight` - The light object
 - `fLerpPeriod` - The lerp period in seconds
-

GetLightLerpPeriod_xpAM

```
float GetLightLerpPeriod_xpAM(  
    object oLight  
);
```

Get the time in seconds for the light to cycle between the color and lerp target color.

Parameters

- `oLight` - The light object

Return Value

The time in seconds for the light to cycle between the color and lerp target color.

SetLightLerpIntensity_xpAM

```
void SetLightLerpIntensity_xpAM(  
    object oLight,  
    float fLerpIntensity  
);
```

Set the intensity (brightness) of the lerp light within its range.

Parameters

- `oLight` - The light object
 - `fLerpIntensity` - The lerp light intensity
-

GetLightLerpIntensity_xpAM

```
float GetLightLerpIntensity_xpAM(  
    object oLight  
);
```

Get the intensity (brightness) of the lerp light within its range.

Parameters

- `oLight` - The light object

Return Value

The intensity (brightness) of the lerp light within its range.

SetLightLerpColor_xpAM

```
void SetLightLerpColor_xpAM(  
    object oLight,  
    string sColorType,  
    string sRGB  
);
```

Set the chosen light lerp color value.

Parameters

- `oLight` - The light object
 - `sColorType` - The [LIGHT_COLOR_TYPE *](#) constant
 - `sRGB` - The RGB string value in the "#FFFFFF" (or #FFFFFFF) format
-

GetLightLerpColor_xpAM

```
string GetLightLerpColor_xpAM(  
    object oLight,  
    string sColorType  
);
```

Get the chosen light lerp color value.

Parameters

- `oLight` - The light object
- `sColorType` - The [LIGHT_COLOR_TYPE *](#) constant

Return Value

The RGB string value of the chosen light lerp color.

Flicker

SetLightIsFlicker_xpAM

```
void SetLightIsFlicker_xpAM(  
    object oLight,  
    int bFlicker  
);
```

Enable or disable flicker for this light.

Parameters

- `oLight` - The light object
 - `bFlicker` - True to enable flicker, false to disable
-

GetLightIsFlicker_xpAM

```
int GetLightIsFlicker_xpAM(  
    object oLight  
);
```

Get the flicker activation status for this light.

Parameters

- `oLight` - The light object

Return Value

True if flicker is enabled for the light, false otherwise.

SetLightFlickerType_xpAM

```
void SetLightFlickerType_xpAM(  
    object oLight,  
    int iFlickerType  
);
```

Set the light flicker type with a [FLICKER_TYPE *](#) value.

Parameters

- `oLight` - The light object
 - `iFlickerType` - The [FLICKER_TYPE *](#) constant
-

GetLightFlickerType_xpAM

```
int GetLightFlickerType_xpAM(  
    object oLight  
);
```

Get the [FLICKER_TYPE *](#) corresponding to the flicker type.

Parameters

- `oLight` - The light object

Return Value

The [FLICKER_TYPE](#) * constant corresponding to the flicker type.

SetLightFlickerVariance_xpAM

```
void SetLightFlickerVariance_xpAM(  
    object oLight,  
    float fVariance  
);
```

Set the flicker variance value.

Parameters

- `oLight` - The light object
 - `fVariance` - The flicker variance value
-

GetLightFlickerVariance_xpAM

```
float GetLightFlickerVariance_xpAM(  
    object oLight  
);
```

Get the flicker variance value.

Parameters

- `oLight` - The light object

Return Value

The flicker variance value.

SetLightFlickerRate_xpAM

```
void SetLightFlickerRate_xpAM(  
    object oLight,  
    float fRate  
);
```

Set the flicker rate for this light.

Parameters

- `oLight` - The light object
- `fRate` - The flicker rate

GetLightFlickerRate_xpAM

```
float GetLightFlickerRate_xpAM(  
    object oLight  
);
```

Get the flicker rate for this light.

Parameters

- `oLight` - The light object

Return Value

The flicker rate for the light.
