

# AMLEngine Insights (Version 3.x)

---

**Josef Prinz**

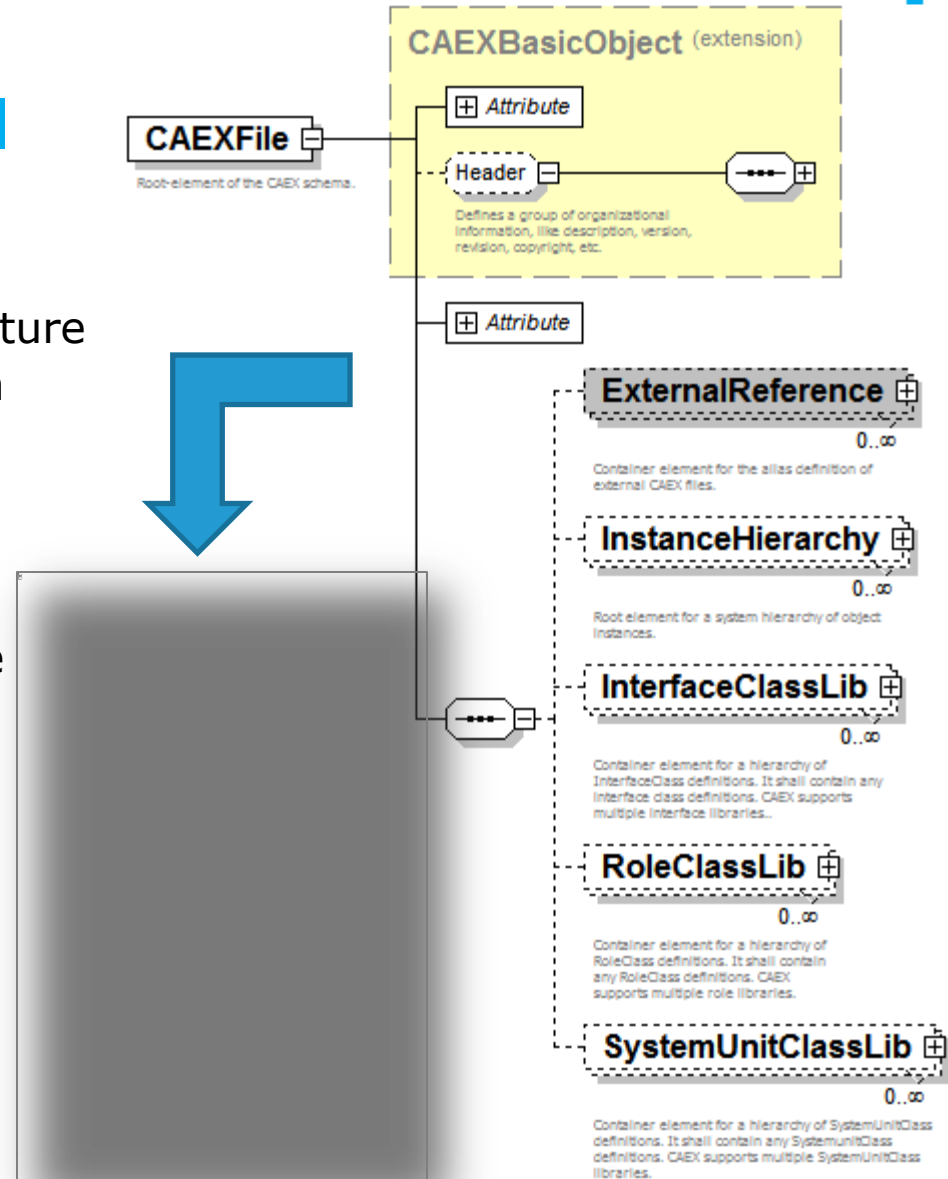
**inpro**

AutomationML PlugFest

Oct. 14th/15th 2015

# AMLEngine Datamodel

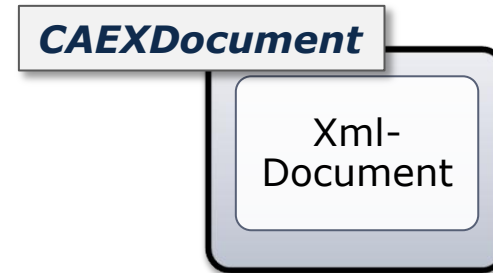
- Representation of the AutomationML-Document structure and the CAEX Schema (version 2.15)
- Each CAEX Schema Element is represented as a Class
- The Datamodel is based on the .NET Classes **XmlDocument** and **XmlNode**
- The Classes are generated with Altova XmlSpy and manually extended



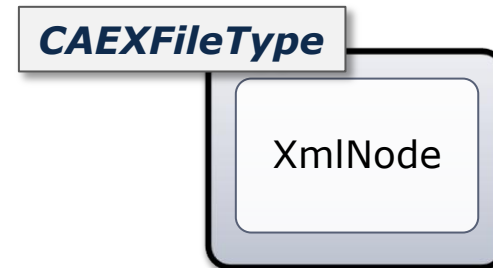
# AMLEngine Datamodel

## Altova Model Generation

- XmlDocument is “wrapped” in the CAEXDocument Class
- The CAEXDocument Class provides methods to load and save AutomationML-Documents and gives the “entry”-Point to the Root “CAEXFile”-Node.
- XmlNodeNodes are “wrapped” in their corresponding Schema-Element Class



**Wrapper for  
“AutomationML-Document”**

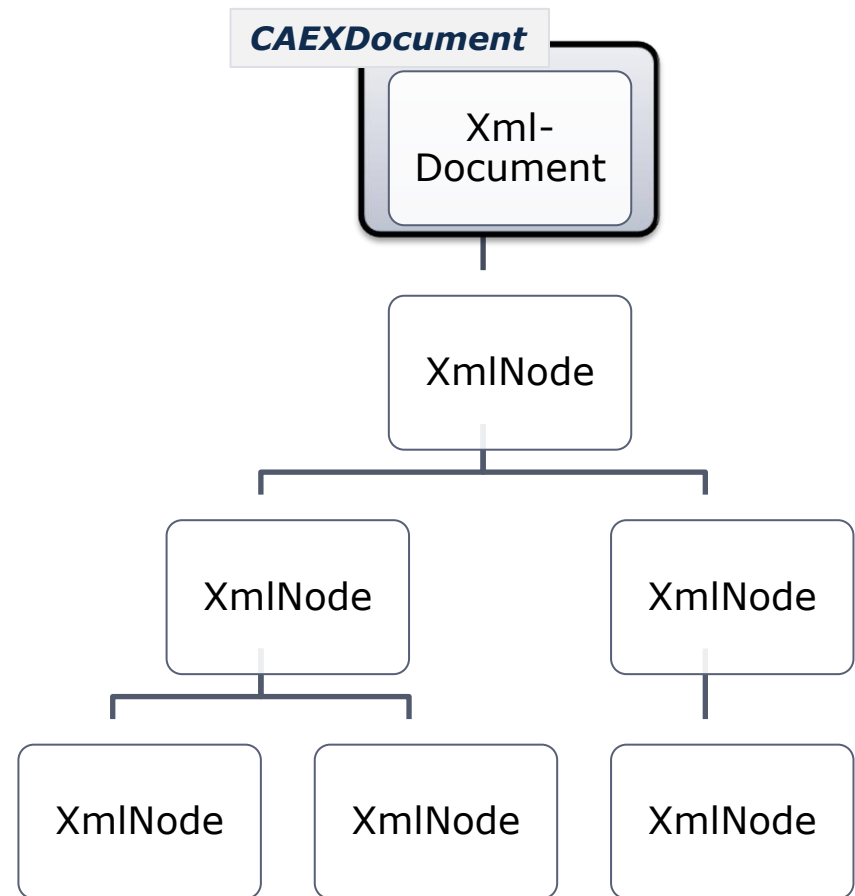


**Wrapper for Schema Element  
“CAEXFile”**

# AMLEngine Datamodel

## AutomationML - Document Representation

- AutomationML-Document is loaded completely in an XmlDocument (XmlNode-Tree)
- Only the CAEXDocument wrapper is created.
- Other CAEXObject-wrappers are created, when an XmlNode is accessed.
- The CAEXDocument wrapper is persistent, as long as the document is in memory.
- Other wrappers are not persistent.

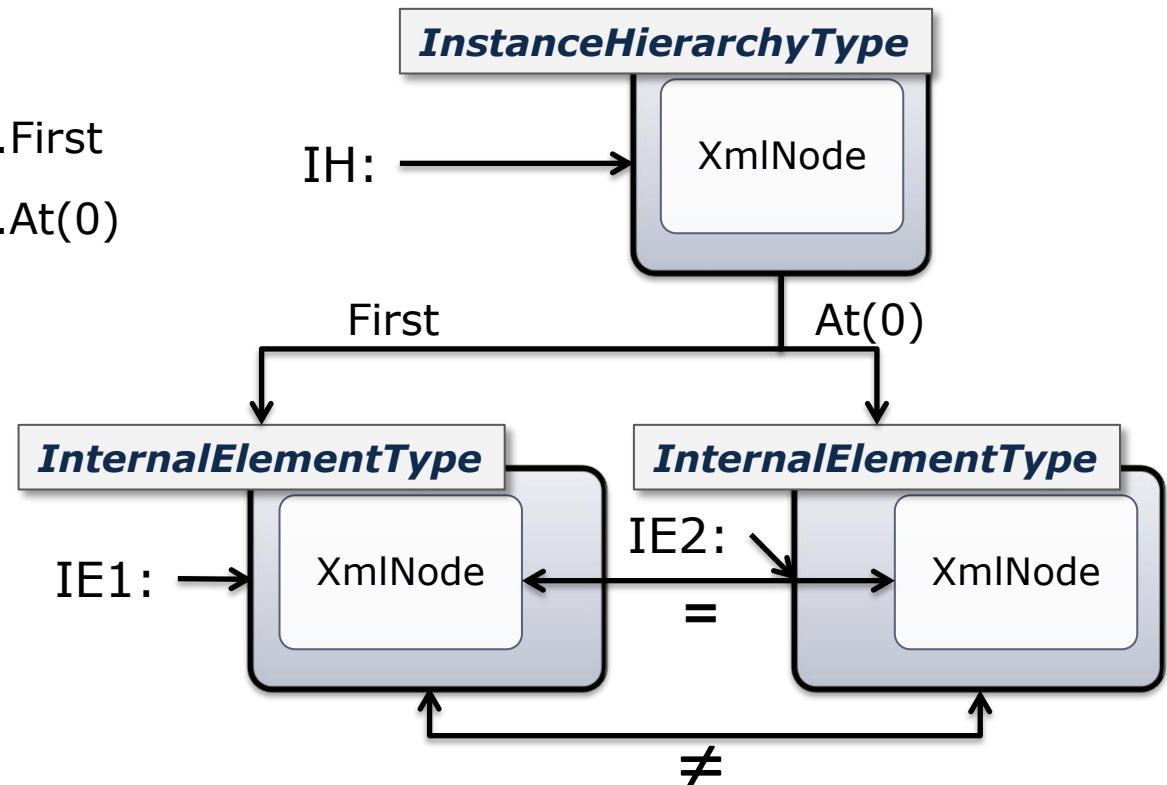


# AMLEngine Datamodel

Example:

- $IE1 = IH.InternalElement.First$
- $IE2 = IH.InternalElement.At(0)$

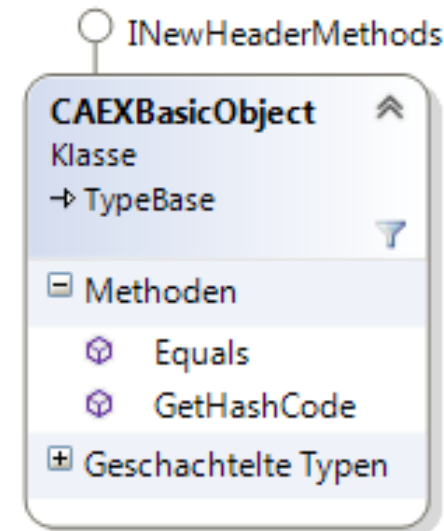
Two wrapped instances of the same XmlNode are **unequal** !



# AMLEngine Datamodel changed in Version 3.0

You can compare wrapped instances of the same XmlNode

- Each wrapper class is derived from the class *CAEXBasicObject*
- *CAEXBasicObject* overrides
  - *Equals*  
*XmlNode.Equals(XmlNode)*
  - *GetHashCode*  
*XmlNode.GetHasCode*
- *Previous Example:*
  - Version 3.x *IE1.Equals(IE2) = true*
  - Version 2.x *IE1.Equals(IE2) = false*



# AMLEngine Datamodel changed in Version 3.0

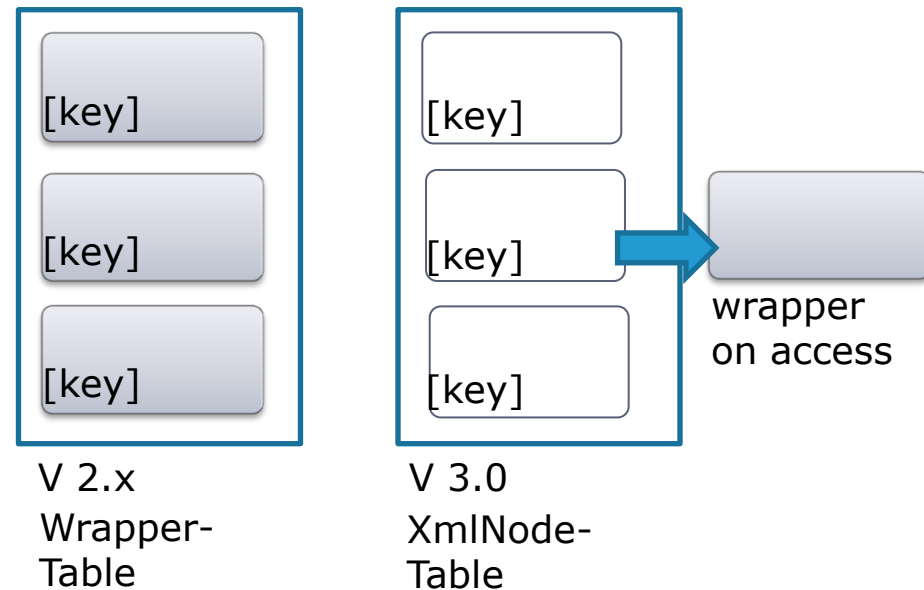
## Implementation of Fast-Access Tables:

- In Version 2.x all Tables store Wrapper instances
- In Version 3.x all Tables store XmlNodes; a wrapper instance is only created, when a node is accessed.

⇒ performance boost for table creation (~ \*50)

⇒ Table creation and update is done with a background worker

⇒ Synchronized when accessed

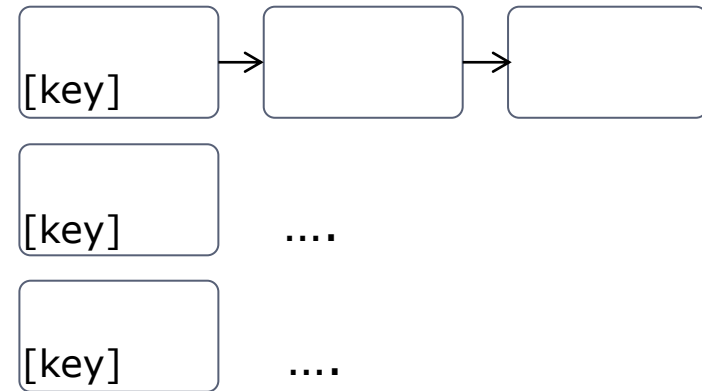


# AMLEngine Datamodel changed in Version 3.0

## Implementation of Fast-Access Tables:

- In Version 2.x the tables contain one instance for one key
- In Version 3.x the tables may contain a list of instances for one key

=> more options for consistency checks and repair functions

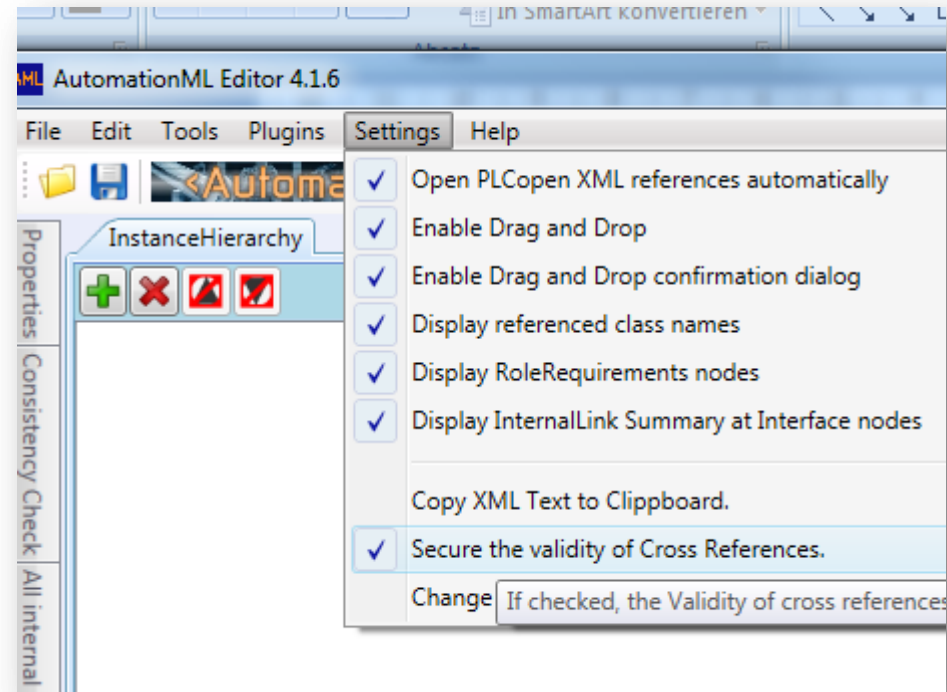




# AMLEngine Datamodel extended in Version 3.0

## Implementation of Fast-Access Tables:

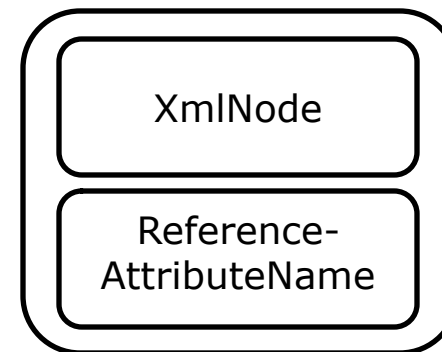
- Version 2.x has one **IDTable** (ID as key) and one **PathTable** (CAEXPath as key)
- Version 3.x has additional **IDRefTable** and **PathRefTable**
  - ⇒ more options for cross reference checks and consistency functions
  - ⇒ Used in AutomationML-Editor 4.x (change a role name of a referenced role or move a referenced role in the hierarchy)



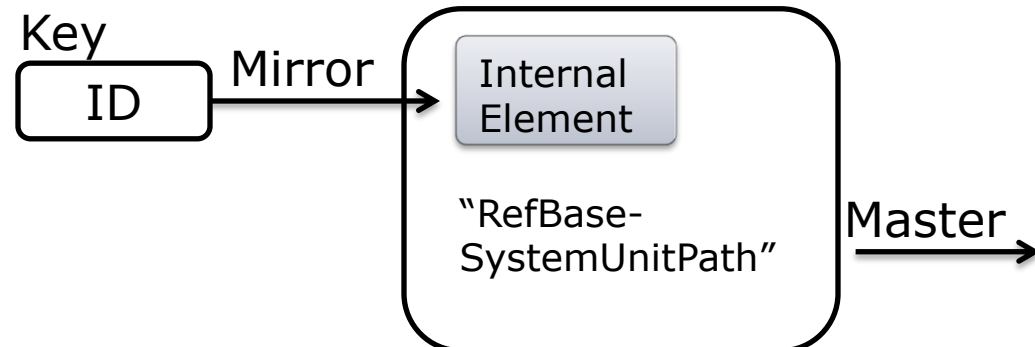
# AMLEngine Datamodel extended in Version 3.0

- Reference tables contain instances of type ReferenceElement
- The Property ReferenceAttributeName defines the CAEXAttribute, which holds the reference value

ReferenceElement



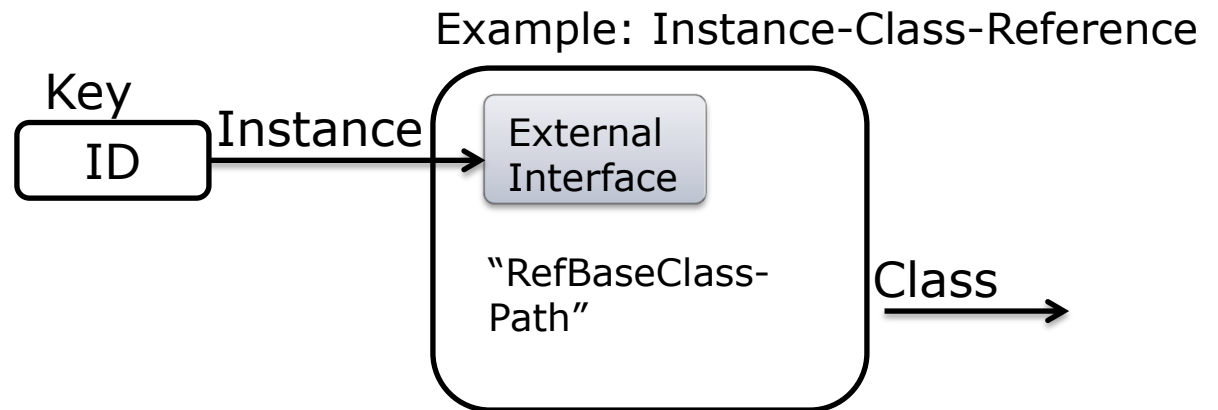
Example: Mirror-Master-Reference



# AMLEngine Datamodel extended in Version 3.0

- **Instance Class Reference in IDRefTable**

- Value of "RefBaseClassPath" is a key to the "PathTable"



# AMLEngine Datamodel changed in Version 3.0

- **Updating tables**

- All tables with one method
- Add and Remove single objects
- Application is responsible for updates
- Application can “inform” the Engine, that it will manage the tables, so that internal methods of the engine can use the tables.





```
CAEXDocument document =  
    CAEXDocument.LoadFromFile("myDoc.aml");  
document.Tables.UpdateAllTables();  
  
// if engine should use the tables  
document.Tables.ManagedByApplication = true;
```

# AMLEngine 3.0 extended Model Validation

- The Validation engine is downward compatible (all 2.x validation methods are still available)
- Validation is done in 4 phases
  1. Table update
  2. Validation
  3. Result interpretation
  4. Repair if it is an option

```
CAEXDocument document =  
    CAEXDocument.LoadFromFile ("myDoc.aml")  
  
// update  
document.Tables.UpdateAllTables();  
  
// validation  
foreach ( var isValidated in document.ValidateAll())  
{  
    // interpretation  
    if (isValidated.AvailableRepairOptions.HasFlag  
        (RepairTypeEnum.GenerateUniqueIDForDoubles))  
    {  
        // repair  
        document.Repair (isValidated,  
            RepairTypeEnum.GenerateUniqueIDForDoubles);  
    }  
}
```

# AMLEngine 3.0 extended Model Validation

	Repair	Repairs the specified validated element. This Method will block, if any Update-Tasks are pending in the <a href="#">Tables</a> Instance.
	ValidateAll	Get Validation Information for all elements, which have validation issues. The ValidationElement provides detailed information about available Repair Options and an additional Information Text. This Method will block, if any Update-Tasks are pending in the <a href="#">Tables</a> Instance.
	ValidateIDsAndNames	Gets all MetaData-Validation Occurrences, all ID-Validation Occurrences and all Path-Validation Occurrences. This Method will block, if any Update-Tasks are pending in the <a href="#">Tables</a> Instance.
	ValidateReferences	Validates all references to CAEX-Objects via an ID-Attribute or a CAEX-Path. This Method will block, if any Update-Tasks are pending in the <a href="#">Tables</a> Instance.

# AMLEngine 3.0 extended Model Validation

## Members

	Member name	Description
	GenerateUniqueIDWhenEmpty	This Flag cause the generation of a unique ID for all Empty IDs
	GenerateUniqueIDForDoubles	This Flag cause the generation of a unique ID for all Double IDs
	GenerateNameWhenEmpty	This Flag cause the renaming of all empty Names with an automatic name generation algorithm
	RemoveEmptyReference	This Flag cause the removal of all references where the Reference Value is empty
	RemoveUnresolvedIDReference	This Flag cause the removal of all ID References where the ID Value has not been resolved
	RemoveUnresolvedPathReference	This Flag cause the removal of all Path References where the Path reference Value has not been resolved
	RenameClassPathForDoubles	This Flag cause the generation of a new name for an element, when it's CAEX-Path is double so that the Path is unique
	CreateWriterHeader	This Flag cause the generation of a writer header
	CreateAutomationMLVersion	This Flag cause the generation of an AutomationML Version header
	AddSchemaLocation	This Flag cause the correction of the Schema Location
	SetSchemaVersion	This Flag cause the correction of the Schema Version
	MergeExternalReference	This Flag cause the external references merged to the current document
	GenerateValidGUID	This Flag cause the generation of a new GUID for Guids with invalid formats
	TryFindInterfaceInInternalLink	This Flag cause the generation of a new InterfaceReference for an InternalLink
	RemoveAlias	This Flag cause the generation of a new reference without the alias
	TryFindClassReference	This Flag cause the generation of a new ClassReference for a Class
	None	The Option None indicates, that nothing should be done

# AMLEngine 3.0 – new optional Parameter

**asFirst = true**  
(**default as in previous versions**)

Element is inserted as the first element of its kind.

**asFirst = false**

Element is inserted as the last element of its kind.

occurs in many methods

## SystemUnitClassType.New\_InternalElement Method

Adding a new InternalElement to this object. If not explicit defined, the InternalElement is added as first InternalElement.

**Namespace:** CAEX\_ClassModel

**Assembly:** CAEX\_ClassModel (in CAEX\_ClassModel.dll)

### Syntax

```
C# VB C++
public InternalElementType New_InternalElement(
    string name,
    bool asFirst
)
```

### Parameters

*name*

Type: String

The name of the new InternalElement

*asFirst*

Type: Boolean

if **true** inserted as first InternalElement (default), otherwise as last







# Vielen Dank für Ihre Aufmerksamkeit!

---

**Haben Sie noch Fragen?**

**inpro**

Innovationsgesellschaft für fortgeschrittene  
Produktionssysteme in der Fahrzeugindustrie mbH

Steinplatz 2

D-10623 Berlin

[www.inpro.de](http://www.inpro.de)