# JS Class Diagram

1. Common
   1-1. protocol
   1-2. entity
2. TypeScript
   2-1. like STL
   2-2. slave system
3. Flex
   3-1. movie
   3-2. file-tree
   3-3. nam-tree

**Protocol Package**

**Invoke Package**

**Invoke**

extends Vector<InvokeParameter>
implements
IDictionary<InvokeParameter>

-listener: string

+Invoke(string)
+Invoke(XML)
+Invoke(string, ... args[]: any)

+has(string) -> boolean
+get(string) -> InvokeParameter

+apply(IProtocol)
+toXML() -> XML

1 1
0 N

**InvokeParameter**

-name: string
-type: string
-value: any

+InvokeParameter(string, any)
+InvokeParameter(string, string, string)

+toXML() -> XML

depends

<<Interface>>

**IProtocol**

**+virtual replyData(Invoke)**
**+virtual sendData(Invoke)**

implements

**ServerConnector**

implements IProtocol

-parent: IProtocol
-socket: WebSocket
-str: string
+opopen: Function

+ServerConnector(IProtocol)
+connect(ip: string, port: number)

+virtual replyData(Invoke)
+virtual sendData(Invoke)

-handleConnect(Event)
-handleReply(MessageEvent)

**Movie Package**

**Application**

implements IProtocol

#socket: ServerConnector
#movie: Movie

+Application(Movie, string, number)
#goAuthority()
#goService()

+virtual replyData(Invoke)
+virtual sendData(Invoke)

contains

**Movie**

implements IProtocol

#application: Application

+Movie()
+setApplication(Application)

+virtual replyData(Invoke)
+virtual sendData(Invoke)

**SubMovie**

implements IProtocol

#parent: IProtocol (Movie||SubMovie)

+SubMovie(IProtocol)
+virtual replyData(Invoke)
+virtual sendData(Invoke)

**Entity Package**

**EntityArray**

extends Vector<IEntity>
implements IEntity,
IDictionary<Entity>

#virtual get CHILD_TAG() -> string

+EntityArray()
+virtual construct(XML)

+virtual toXML() -> XML

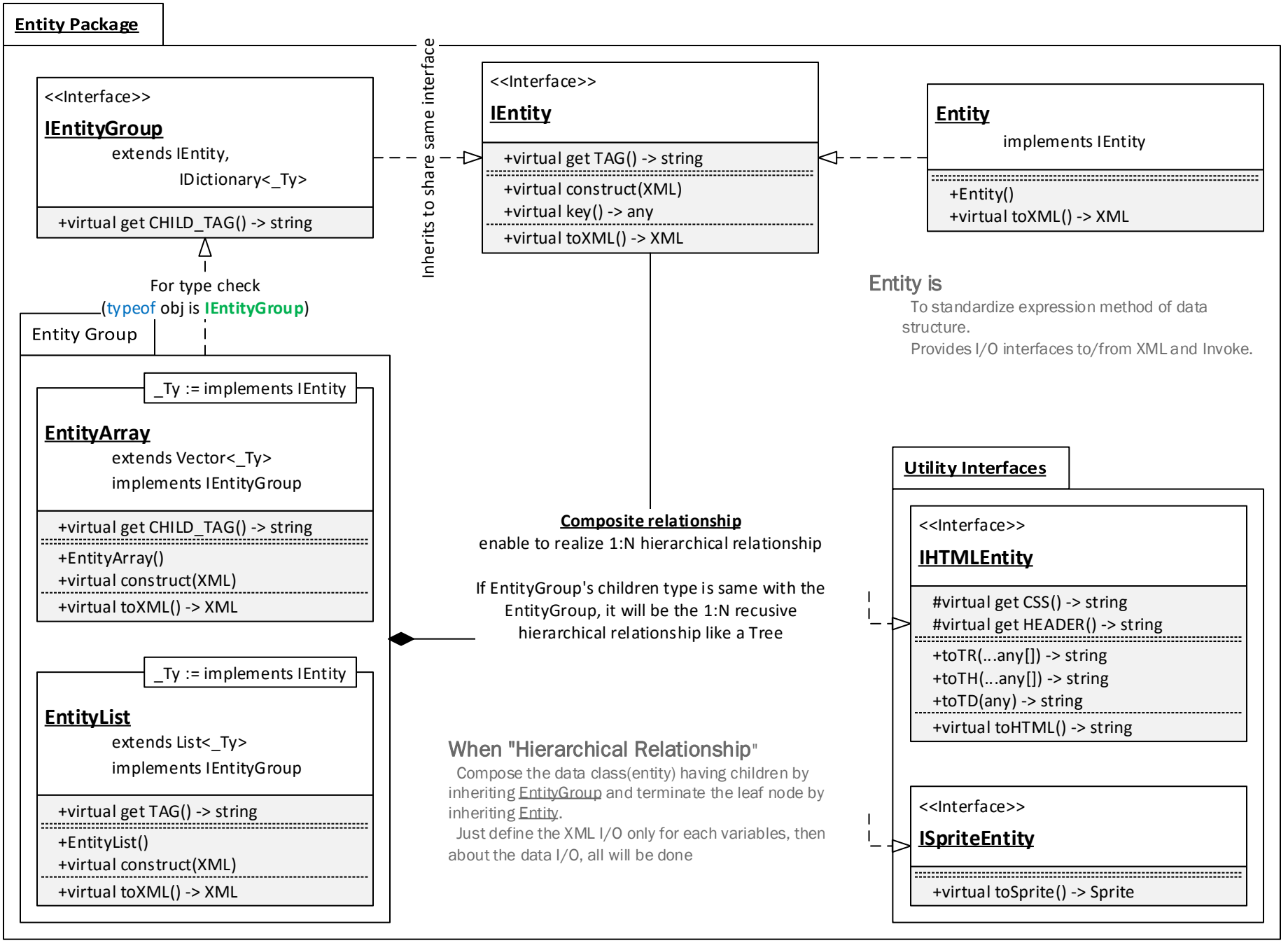
**Entity**

implements IEntity

#virtual get TAG() -> string

+Entity()
+virtual construct(XML)

+virtual toXML() -> XML


Entity Package
<<Interface>>
IEntityGroup
extends IEntity,
IDictionary<_Ty>
+virtual get CHILD_TAG() -> string
Inherits to share same interface
<<Interface>>
IEntity
+virtual get TAG() -> string
+virtual construct(XML)
+virtual key() -> any
+virtual toXML() -> XML
Entity
implements IEntity
+Entity()
+virtual toXML() -> XML
Entity is
To standardize expression method of data structure.
Provides I/O interfaces to/from XML and Invoke.
For type check
(typeof obj is IEntityGroup)
Entity Group
_Ty := implements IEntity
EntityArray
extends Vector<_Ty>
implements IEntityGroup
+virtual get CHILD_TAG() -> string
+EntityArray()
+virtual construct(XML)
+virtual toXML() -> XML
_Ty := implements IEntity
EntityList
extends List<_Ty>
implements IEntityGroup
+virtual get TAG() -> string
+EntityList()
+virtual construct(XML)
+virtual toXML() -> XML
Composite relationship
enable to realize 1:N hierarchical relationship
If EntityGroup's children type is same with the EntityGroup, it will be the 1:N recusive hierarchical relationship like a Tree
When "Hierarchical Relationship"
Compose the data class(entity) having children by inheriting EntityGroup and terminate the leaf node by inheriting Entity.
Just define the XML I/O only for each variables, then about the data I/O, all will be done
Utility Interfaces
<<Interface>>
IHTMLEntity
#virtual get CSS() -> string
#virtual get HEADER() -> string
+toTR(...any[]) -> string
+toTH(...any[]) -> string
+toTD(any) -> string
+virtual toHTML() -> string
<<Interface>>
ISpriteEntity
+virtual toSprite() -> Sprite

**TypeScript's library**

**Like STL**

_Ty

**Vector**

implements Array<_Ty>

+length: double
+[n: number]: _Ty

+Vector(... args: _Ty[])
+{methods of Array}

_Ty1, _Ty2

**Pair**

+first: _Ty1
+second: _Ty2

+Pair(_Ty1, _Ty2)
+equals(Pair)

+toString() -> string

_Kty, _Ty

<<Interface>>

**IMap**

+has(_Kty) -> boolean
+get(_Kty) -> _Ty

+set(_Kty, _Ty)

_Kty, _Ty

**Map**

implements IMap<_Kty, _Ty>

-data_: Vector<Pair<_Kty, _Ty>>

+Map()
+data() -> Vector<Pair<_Kty, _Ty>>
+size() -> number

+has(_Kty) -> boolean
+get(_Kty) -> _Ty
+set(_Kty, _Ty)

+find(_Kty) -> MapIterator<_Kty, _Ty>
+begin() -> MapIterator<_Kty, _Ty>
+end() -> MapIterator<_Kty, _Ty>

+erase(_Kty)
+clear()

+equals(Map) -> boolean
+toString() -> string

_Kty, _Ty

**MapIterator**

-map: Map<_Kty, _Ty>
-index: number

+MapIterator(Map<_Kty, _Ty>, number)
+prev() -> MapIterator<_Kty, _Ty>
+next() -> MapIterator<_Kty, _Ty>

+set first(_Kty)
+set second(_Kty)
+get first() -> _Kty
+get second() -> _Ty

+equals(MapIterator<_Kty, _Ty>) -> bool
+toString() -> string

_Ty

<<Interface>>

**IDictionary**

extends IMap<string, _Ty>

_Ty

**Dictionary**

extends Map<string, _Ty>
implements IDictionary<_Ty>

**XML Package**

**XML**

extends Dictionary<XMLList>,
implements IDictionary<XMLList>

-tag: string
-value: any
-properties: Dictionary<any>

+XML(string := "")
-parse(string)
-parseTag(string)
-parseProperties(string)
-parseValue(string) -> Pair<string, bool>
-parseChildren(string)

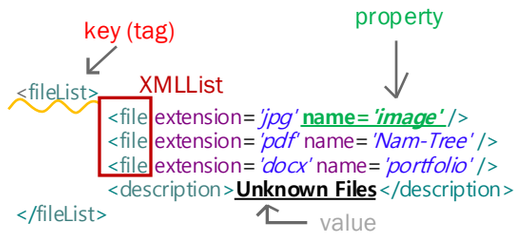
+hasProperty(string) -> boolean
+getProperty(string) -> any
+setProperty(string, any)

+toString(number) -> string

**XMLList**

extends Vector<XML>

+XMLList(... args: XML[])
+toString(number) -> string



**Flex - BOUNDARY**

<<Interface>>

**IProtocol**

+virtual sendData(Invoke)
+virtual replyData(Invoke)

**TitleWindow**

extends spark.TitleWindow
implements IProtocol

#window: Window
#creationFlag: Boolean

+TitleWindow(Window)
#virtual creationCompleted(FlexEvent)

+virtual sendData(Invoke)
**+virtual replyData(Invoke)**

+<<static>> createPopUp
(Window, Class) -> TitleWindow

opened by

<<Abstract>>

**Window**

extends spark.Application
implements IProtocol

#socket: CPPSocket
#topMenu: TopMenu
#movie: Movie
#creationFlag: Boolean

+Window()
#virtual creationCompleted(FlexEvent)
**+virtual createTopMenu() -> TopMenu**
**+virtual createMovie() -> Movie**

#constructVBox()
#constructTopMenu()
#constructMovie()

#goServiceInfo()
#goConnect(String, int)
#goAuthoirty()
#handleServerInfo(Event)
#handleConnect(Event)
#handleAuthority(int, Boolean)

+virtual sendData(Invoke)
**+virtual replyData(Invoke)**
-handleReply(CPPSocketEvent)

+goWindow(String)
+goPopUp(Class)
+goExport(String)
+goScreenshot()
+goClose()

contains in
upper-side

<<Abstract>>

**TopMenu**

extends mx.TabNavigator

#window: Window

+TopMenu(Window)
+setAuthority(int)

contained

**MenuButton**

extends spark.VGroup

+set label(String)
+set image(String)
+set source(String)

contains
in
down-side

<<Abstract>>

**Movie**

extends spark.Group
implements IProtocol

#window: Window
#creationFlag: Boolean

+Movie(Window)
#virtual creationCompleted(FlexEvent)

+virtual sendData(Invoke)
**+virtual replyData(Invoke)**

as a sub-part

<<Abstract>>

**SubMovie**

extends spark.Group
implements IProtocol

#patientMovie: IInvoke
#titleWindow: SubMovieTitleWindow
#creationFlag: Boolean

+SubMovie(IInvoke)
#virtual creationCompleted(FlexEvent)

+virtual sendData(Invoke)
**+virtual replyData(Invoke)**

#virtual goPopUp(Class) ->
SubMovieTitleWindow

As a sub-part
recursively

creatss

**SubMovieTitleWindow**

extends spark.TitleWindow
implements IProtocol

#subMovie: SubMovie
#creationFlag: Boolean

+SubMovieTitleWindow(SubMovie)
#vrtual creationCompleted(FlexEvent)

+virtual sendData(Invoke)
**+virtual replyData(Invoke)**

+<<static>> createPopUp(SubMovie,
Class) -> SubMovieTitleWindow

Flex - FileTree

File-Tree Entity Package

**FTInstance**
implements IEntity

+virtual get key() -> String

#parent: FTFolder
#name: String
#comment: String

**+virtual get label() -> String**
**+virtual get path() -> String**

**+virtual construct(XML)**
**+virtual toXML() -> XML**

**FTFolder**
extends EntityArray
implments FTInstance

+v. get CHILD_TAG() -> String = "file"
#factory: FTManager

+FTFolder(FTManager, FTFolder)
+virtual construct(XML)
#virtual createChild(XML) -> Entity

+virtual toXML() -> XML

**FTFile**
extends Entity
implements FTInstance

+virtual get label() -> String
#extension: String

+FTFile(FTFolder)
**+virtual construct(XML)**

**+virtual toXML() -> XML**

File-Tree Boundaries

**FTManager**
extends SubMovie

#top Folder: FTFolder
#fileMap: Dictionary<IFTFile>
#creationWindow: FTCreationWindow
**+virtual get selectedFile() -> IFTFile**
**#v. get FileCreationWindow() -> Class**

**#virtual creationCompleted(FlexEvent)**
**+virtual createFile(XML) -> IFTFile**
**#virtual createWindow(String)**

**#virtual handleIcon(IFTTile) -> Class**
#isCreatable(String) -> Boolean
#goDelete(MouseEvent)
#handleDoubleClicked(MouseEvent)

**+virtual replyData(Invoke)**
#setFileTree(XML)
#setFile(XML)
#notifyFileCreationCompleted()

1 — 1 Creates Titlewindow 0 — 1

**FTCreationWindow**
extends SubMovieTitleWindow

#file: IFTFile

+FTCreationWindow(FTManager, IFTFile)
**#virtual creationCompleted(FlexEvent)**

**#virtual goInit(MouseEvent)**
-goArchive(MouseEvent)
**+virtual replyData(Invoke)**

**FolderCreationWindow**
extends FTCreationWindow

+FolderCreationWindow
(FTManager, IFTFile)
**#virtual creationCompleted(FlexEvent)**

**TextFileCreationWindow**
extends FTCreationWindow

+TextFileCreationWindow
(FTManager, IFTFile)
**#virtual creationCompleted(FlexEvent)**

**ConfigurationFTManager**
extends FTManager

-memberName

#virtual creationCompleted(FlexEvent)
#virtual handleDoubleClicked
(MouseEvent)

**ConfigurationExplorerWindow**
extends TitleWindow

#manager: NTManager
#target: IEntity

#virtual creationCompleted(FlexEvent)
+virtual replyData(Invoke)

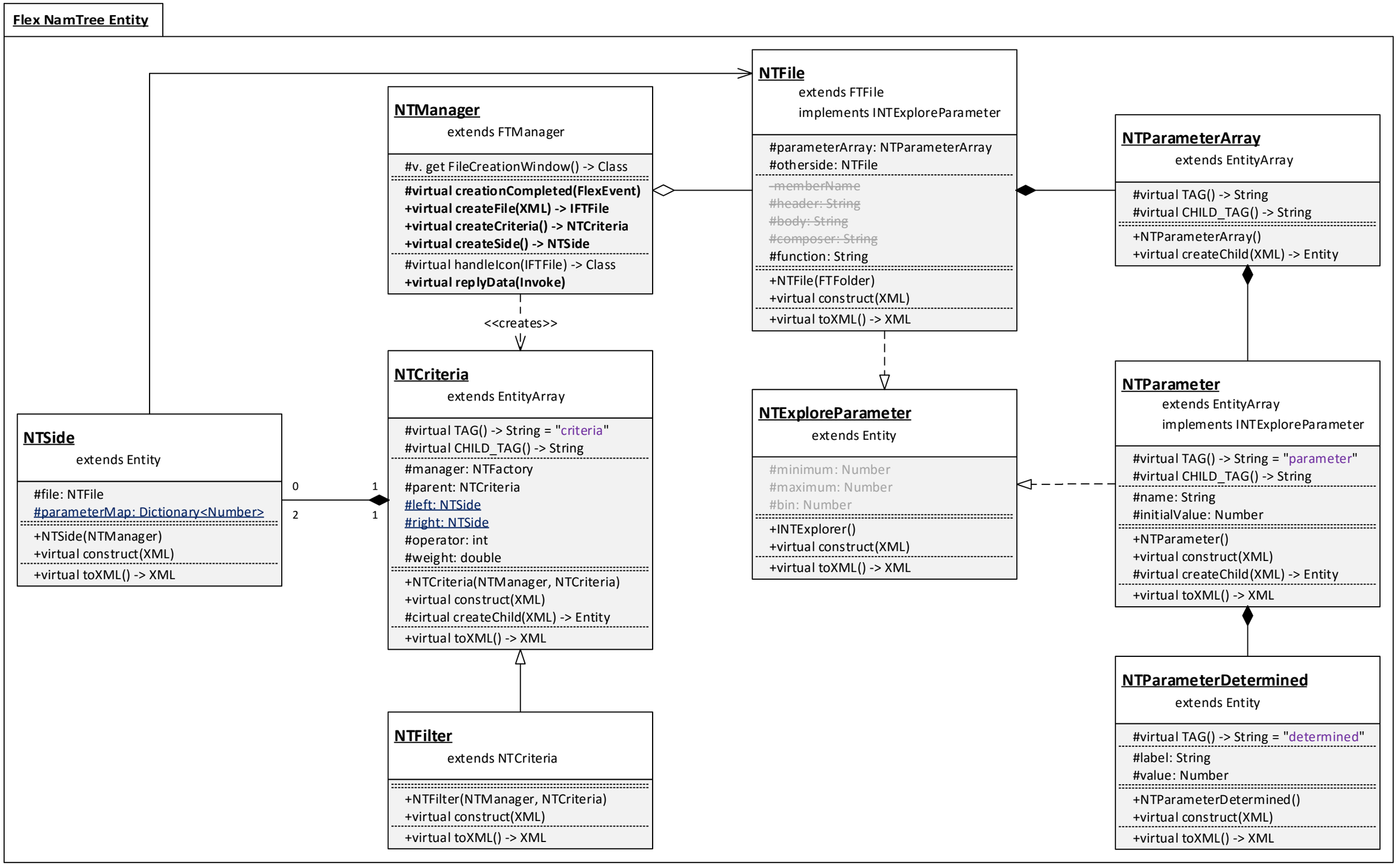
**#virtual handleFileChanged(ListEvent)**
**#virtual goDetermine(MouseEvent)**

**ConfigurationOpenWindow**
extends ConfigurationExplorerWindow

#virtual creationCompleted(FlexEvent)
#virtual goDetermine(MouseEvent)

**ConfigurationSaveWindow**
extends ConfigurationExplorerWindow

#virtual creationCompleted(FlexEvent)
#virtual goDetermine(MouseEvent)


Flex NamTree Entity
NTFile
extends FTFile
implements INTExploreParameter
#parameterArray: NTParameterArray
#otherside: NTFile
-memberName
#header: String
#body: String
#composer: String
#function: String
+NTFile(FTFolder)
+virtual construct(XML)
+virtual toXML() -> XML
NTManager
extends FTManager
#v. get FileCreationWindow() -> Class
#virtual creationCompleted(FlexEvent)
+virtual createFile(XML) -> IFTFile
+virtual createCriteria() -> NTCriteria
+virtual createSide() -> NTSide
#virtual handleIcon(IFTFile) -> Class
+virtual replyData(Invoke)
<<creates>>
NTParameterArray
extends EntityArray
#virtual TAG() -> String
#virtual CHILD_TAG() -> String
+NTParameterArray()
+virtual createChild(XML) -> Entity
NTCriteria
extends EntityArray
#virtual TAG() -> String = "criteria"
#virtual CHILD_TAG() -> String
#manager: NTFactory
#parent: NTCriteria
#left: NTSide
#right: NTSide
#operator: int
#weight: double
+NTCriteria(NTManager, NTCriteria)
+virtual construct(XML)
#cirtual createChild(XML) -> Entity
+virtual toXML() -> XML
NTExploreParameter
extends Entity
#minimum: Number
#maximum: Number
#bin: Number
+INTExplorer()
+virtual construct(XML)
+virtual toXML() -> XML
NTParameter
extends EntityArray
implements INTExploreParameter
#virtual TAG() -> String = "parameter"
#virtual CHILD_TAG() -> String
#name: String
#initialValue: Number
+NTParameter()
+virtual construct(XML)
#virtual createChild(XML) -> Entity
+virtual toXML() -> XML
NTSide
extends Entity
#file: NTFile
#parameterMap: Dictionary<Number>
+NTSide(NTManager)
+virtual construct(XML)
+virtual toXML() -> XML
0
2
1
1
NTFilter
extends NTCriteria
+NTFilter(NTManager, NTCriteria)
+virtual construct(XML)
+virtual toXML() -> XML
NTParameterDetermined
extends Entity
#virtual TAG() -> String = "determined"
#label: String
#value: Number
+NTParameterDetermined()
+virtual construct(XML)
+virtual toXML() -> XML

**Flex - NamTree UI**

## NTManager

extends FTManager

#v. get FileCreationWindow() -> Class

**#virtual creationCompleted(FlexEvent)**
**+virtual createFile(XML) -> IFTFile**
**+virtual createCriteria() -> NTCriteria**
**+virtual createSide() -> NTSide**

#virtual handleIcon(IFTFile) -> Class
**+virtual replyData(Invoke)**

1 — 1 — creates window — 0 — 1

## NTCreationWindow

extends SubMovieTitleWindow

#get parameters() -> NTParameterArray
#get determineds() -> NTParameter

+NTCreationWindow(FTManager, IFile)
**#virtual creationCompleted(FlexEvent)**

**#virtual goInit(MouseEvent)**
**+virtual replyData(Invoke)**

contains

## NTFileParameterGrid

extends mx.HGroup

#window: NTCreationWindow
#parameterGrid: mx.AdvancedDataGrid
#determinedGrid: mx.AdvancedDataGrid

#creationCompleted(FlexEvent)
-add(MouseEvent)
-addDetermined(MouseEvent)

-remove(MouseEvent)
-removeDetermined(MouseEvent)

1 — 1

be contained in right side

1 — 1

## NTMovie

extends SubMovie

**#v. get criteriaGrid() -> NTCriteriaGrid**
#manager: NTManager
#criteriaGrids: Vector<NTCriteriaGrid>
#leftSideContainer: NTSideContainer
#rightSideContainer: NTSideContainer

**#virtual creationCompleted(FlexEvent)**
**#virtual constructUI()**

**#virtual createManager() -> NTManager**
**#virtual createCriteriaGrid()**
**-> NTCriteriaGrid**
**#virtual createSideContainer()**
**-> NTSideContainer**

-goApplyFile(enum := DIRECTION)
-itemGridChanged(ListEvent)

1 — 1 — Sometimes can be multiple — 1 — 1

## NTCriteriaGrid

extends mx.AdvancedDataGrid

#manager: NTManager
#topCriteria: NTCriteria
#changeHandler: Function
+get selectedCriteria() -> NTCriteria

**#virtual creationCompleted(FlexEvent)**

-initialize(MouseEvent)
-goImport(MouseEvent, NTCriteria)
-goExport(MouseEvent, NTCriteria)

-addFilter(MouseEvent)
-addCriteria(MouseEvent)
-removeCriteria(MouseEvent)

1 — 1

contains in bottom side

1 — 2

## NTSideContainer

extends mx.VGroup

#direction: enum {LEFT, RIGHT}
#parameterGrid: mx.AdvancedDataGrid
#side: NTSide

**#virtual creationCompleted(FlexEvent)**
+refresh()
+setSide(NTSide)

-valueChanged(Event)