

RageBMLNet

Generated by Doxygen 1.8.12

Contents

| | | |
|----------|---|----------|
| 1 | Namespace Index | 1 |
| 1.1 | Packages | 1 |
| 2 | Hierarchical Index | 3 |
| 2.1 | Class Hierarchy | 3 |
| 3 | Class Index | 5 |
| 3.1 | Class List | 5 |
| 4 | Namespace Documentation | 9 |
| 4.1 | AssetManagerPackage Namespace Reference | 9 |
| 4.1.1 | Detailed Description | 9 |
| 4.2 | AssetPackage Namespace Reference | 10 |
| 4.2.1 | Enumeration Type Documentation | 11 |
| 4.2.1.1 | LogLevel | 11 |
| 4.2.1.2 | Severity | 11 |
| 4.3 | BMLNet Namespace Reference | 11 |

| | | |
|----------|--|-----------|
| 5 | Class Documentation | 13 |
| 5.1 | AssetManagerPackage.AssetManager Class Reference | 13 |
| 5.1.1 | Detailed Description | 14 |
| 5.1.2 | Member Function Documentation | 14 |
| 5.1.2.1 | findAssetByClass() | 14 |
| 5.1.2.2 | findAssetById() | 14 |
| 5.1.2.3 | findAssetsByClass() | 14 |
| 5.1.2.4 | getInterface< T >() | 15 |
| 5.1.2.5 | Log() [1/2] | 15 |
| 5.1.2.6 | Log() [2/2] | 15 |
| 5.1.2.7 | registerAssetInstance() | 16 |
| 5.1.3 | Property Documentation | 16 |
| 5.1.3.1 | Bridge | 16 |
| 5.1.3.2 | Instance | 16 |
| 5.1.3.3 | VersionAndDependenciesReport | 16 |
| 5.2 | AssetPackage.BaseAsset Class Reference | 17 |
| 5.2.1 | Detailed Description | 18 |
| 5.2.2 | Constructor & Destructor Documentation | 18 |
| 5.2.2.1 | BaseAsset() [1/2] | 18 |
| 5.2.2.2 | BaseAsset() [2/2] | 18 |
| 5.2.3 | Member Function Documentation | 19 |
| 5.2.3.1 | GetEmbeddedResource() | 19 |
| 5.2.3.2 | getInterface< T >() | 19 |
| 5.2.3.3 | LoadDefaultSettings() | 19 |
| 5.2.3.4 | LoadSettings() | 20 |
| 5.2.3.5 | Log() [1/2] | 20 |
| 5.2.3.6 | Log() [2/2] | 20 |
| 5.2.3.7 | SaveDefaultSettings() | 20 |
| 5.2.3.8 | SaveSettings() | 21 |
| 5.2.3.9 | SettingsFromXml() | 21 |

| | | |
|----------|---|----|
| 5.2.3.10 | SettingsToXml() | 21 |
| 5.2.4 | Property Documentation | 22 |
| 5.2.4.1 | Bridge | 22 |
| 5.2.4.2 | Class | 22 |
| 5.2.4.3 | Dependencies | 22 |
| 5.2.4.4 | hasSettings | 22 |
| 5.2.4.5 | Id | 22 |
| 5.2.4.6 | Maturity | 23 |
| 5.2.4.7 | Settings | 23 |
| 5.2.4.8 | Version | 23 |
| 5.2.4.9 | VersionInfo | 23 |
| 5.3 | AssetPackage.BaseSettings Class Reference | 23 |
| 5.3.1 | Detailed Description | 24 |
| 5.3.2 | Constructor & Destructor Documentation | 24 |
| 5.3.2.1 | BaseSettings() | 24 |
| 5.3.3 | Member Function Documentation | 24 |
| 5.3.3.1 | UpdateDefaultValues() | 24 |
| 5.4 | BMLNet.BMLBehavior Class Reference | 24 |
| 5.4.1 | Detailed Description | 25 |
| 5.4.2 | Constructor & Destructor Documentation | 25 |
| 5.4.2.1 | BMLBehavior() | 25 |
| 5.4.3 | Member Function Documentation | 26 |
| 5.4.3.1 | Parse() | 26 |
| 5.5 | BMLNet.BMLBlock Class Reference | 26 |
| 5.5.1 | Detailed Description | 28 |
| 5.5.2 | Constructor & Destructor Documentation | 28 |
| 5.5.2.1 | BMLBlock() | 28 |
| 5.5.3 | Member Function Documentation | 28 |
| 5.5.3.1 | Parse() | 28 |
| 5.5.3.2 | TryParseAttribute< T >() | 28 |

| | | |
|---------|--|----|
| 5.5.3.3 | TryParseSyncPoint() | 29 |
| 5.5.4 | Member Data Documentation | 30 |
| 5.5.4.1 | id | 30 |
| 5.5.4.2 | parentBml | 30 |
| 5.5.4.3 | syncPoints | 30 |
| 5.6 | BMLNet.BMLBml Class Reference | 30 |
| 5.6.1 | Member Enumeration Documentation | 31 |
| 5.6.1.1 | Composition | 31 |
| 5.6.2 | Constructor & Destructor Documentation | 31 |
| 5.6.2.1 | BMLBml() | 31 |
| 5.6.3 | Member Function Documentation | 31 |
| 5.6.3.1 | Parse() | 31 |
| 5.6.4 | Member Data Documentation | 32 |
| 5.6.4.1 | characterId | 32 |
| 5.6.4.2 | composition | 32 |
| 5.7 | BMLNet.BMLFace Class Reference | 32 |
| 5.7.1 | Detailed Description | 33 |
| 5.7.2 | Constructor & Destructor Documentation | 33 |
| 5.7.2.1 | BMLFace() | 33 |
| 5.7.3 | Member Function Documentation | 33 |
| 5.7.3.1 | Parse() | 33 |
| 5.7.4 | Member Data Documentation | 33 |
| 5.7.4.1 | amount | 33 |
| 5.7.4.2 | overshoot | 34 |
| 5.8 | BMLNet.BMLFaceFacs Class Reference | 34 |
| 5.8.1 | Detailed Description | 35 |
| 5.8.2 | Member Enumeration Documentation | 35 |
| 5.8.2.1 | Side | 35 |
| 5.8.3 | Constructor & Destructor Documentation | 35 |
| 5.8.3.1 | BMLFaceFacs() | 35 |

| | | |
|----------|--|----|
| 5.8.4 | Member Data Documentation | 35 |
| 5.8.4.1 | au | 35 |
| 5.8.4.2 | side | 35 |
| 5.9 | BMLNet.BMLFaceLexeme Class Reference | 36 |
| 5.9.1 | Detailed Description | 36 |
| 5.9.2 | Member Enumeration Documentation | 36 |
| 5.9.2.1 | Lexeme | 36 |
| 5.9.3 | Constructor & Destructor Documentation | 37 |
| 5.9.3.1 | BMLFaceLexeme() | 37 |
| 5.9.4 | Member Function Documentation | 37 |
| 5.9.4.1 | Parse() | 37 |
| 5.9.5 | Member Data Documentation | 37 |
| 5.9.5.1 | lexeme | 37 |
| 5.10 | BMLNet.BMLFaceShift Class Reference | 37 |
| 5.10.1 | Detailed Description | 38 |
| 5.11 | BMLNet.BMLFeedback Class Reference | 38 |
| 5.12 | BMLNet.BMLGaze Class Reference | 38 |
| 5.12.1 | Detailed Description | 39 |
| 5.12.2 | Member Enumeration Documentation | 39 |
| 5.12.2.1 | Direction | 39 |
| 5.12.2.2 | Influence | 39 |
| 5.12.3 | Constructor & Destructor Documentation | 39 |
| 5.12.3.1 | BMLGaze() | 39 |
| 5.12.4 | Member Function Documentation | 39 |
| 5.12.4.1 | Parse() | 39 |
| 5.12.5 | Member Data Documentation | 40 |
| 5.12.5.1 | influence | 40 |
| 5.12.5.2 | offsetAngle | 40 |
| 5.12.5.3 | offsetDirection | 40 |
| 5.12.5.4 | target | 40 |

| | |
|---|----|
| 5.13 BMLNet.BMLGazeShift Class Reference | 41 |
| 5.13.1 Detailed Description | 41 |
| 5.13.2 Constructor & Destructor Documentation | 41 |
| 5.13.2.1 BMLGazeShift() | 41 |
| 5.13.3 Member Function Documentation | 41 |
| 5.13.3.1 Parse() | 41 |
| 5.14 BMLNet.BMLGesture Class Reference | 42 |
| 5.14.1 Detailed Description | 43 |
| 5.14.2 Member Enumeration Documentation | 43 |
| 5.14.2.1 Lexeme | 43 |
| 5.14.2.2 Mode | 43 |
| 5.14.3 Constructor & Destructor Documentation | 43 |
| 5.14.3.1 BMLGesture() | 43 |
| 5.14.4 Member Function Documentation | 43 |
| 5.14.4.1 Parse() | 43 |
| 5.14.5 Member Data Documentation | 44 |
| 5.14.5.1 lexeme | 44 |
| 5.14.5.2 mode | 44 |
| 5.15 BMLNet.BMLHead Class Reference | 44 |
| 5.15.1 Detailed Description | 45 |
| 5.15.2 Member Enumeration Documentation | 45 |
| 5.15.2.1 Lexeme | 45 |
| 5.15.3 Constructor & Destructor Documentation | 45 |
| 5.15.3.1 BMLHead() | 45 |
| 5.15.4 Member Function Documentation | 45 |
| 5.15.4.1 Parse() | 45 |
| 5.15.5 Member Data Documentation | 46 |
| 5.15.5.1 amount | 46 |
| 5.15.5.2 lexeme | 46 |
| 5.15.5.3 repetition | 46 |

| | |
|---|----|
| 5.16 BMLNet.BMLHeadDirectionShift Class Reference | 46 |
| 5.16.1 Detailed Description | 47 |
| 5.16.2 Constructor & Destructor Documentation | 47 |
| 5.16.2.1 BMLHeadDirectionShift() | 47 |
| 5.16.3 Member Function Documentation | 47 |
| 5.16.3.1 Parse() | 47 |
| 5.16.4 Member Data Documentation | 47 |
| 5.16.4.1 target | 48 |
| 5.17 BMLNet.BMLLocomotion Class Reference | 48 |
| 5.17.1 Detailed Description | 48 |
| 5.17.2 Constructor & Destructor Documentation | 49 |
| 5.17.2.1 BMLLocomotion() | 49 |
| 5.17.3 Member Function Documentation | 49 |
| 5.17.3.1 Parse() | 49 |
| 5.17.4 Member Data Documentation | 49 |
| 5.17.4.1 manner | 49 |
| 5.17.4.2 target | 49 |
| 5.18 BMLNet.BMLPointing Class Reference | 50 |
| 5.18.1 Detailed Description | 50 |
| 5.18.2 Member Enumeration Documentation | 50 |
| 5.18.2.1 Mode | 50 |
| 5.18.3 Constructor & Destructor Documentation | 51 |
| 5.18.3.1 BMLPointing() | 51 |
| 5.18.4 Member Function Documentation | 51 |
| 5.18.4.1 Parse() | 51 |
| 5.18.5 Member Data Documentation | 51 |
| 5.18.5.1 mode | 51 |
| 5.18.5.2 target | 51 |
| 5.19 BMLNet.BMLPose Class Reference | 52 |
| 5.19.1 Detailed Description | 53 |

| | | |
|----------|--|----|
| 5.19.2 | Member Enumeration Documentation | 53 |
| 5.19.2.1 | Lexeme | 53 |
| 5.19.2.2 | Part | 53 |
| 5.19.3 | Constructor & Destructor Documentation | 53 |
| 5.19.3.1 | BMLPose() | 53 |
| 5.19.4 | Member Function Documentation | 53 |
| 5.19.4.1 | Parse() | 53 |
| 5.19.5 | Member Data Documentation | 54 |
| 5.19.5.1 | lexeme | 54 |
| 5.19.5.2 | part | 54 |
| 5.20 | BMLNet.BMLPosture Class Reference | 54 |
| 5.20.1 | Detailed Description | 55 |
| 5.20.2 | Constructor & Destructor Documentation | 55 |
| 5.20.2.1 | BMLPosture() | 55 |
| 5.20.3 | Member Function Documentation | 55 |
| 5.20.3.1 | Parse() | 55 |
| 5.21 | BMLNet.BMLPostureShift Class Reference | 55 |
| 5.21.1 | Detailed Description | 56 |
| 5.21.2 | Constructor & Destructor Documentation | 56 |
| 5.21.2.1 | BMLPostureShift() | 56 |
| 5.21.3 | Member Function Documentation | 56 |
| 5.21.3.1 | Parse() | 56 |
| 5.22 | BMLNet.BMLSpeech Class Reference | 56 |
| 5.22.1 | Detailed Description | 57 |
| 5.22.2 | Constructor & Destructor Documentation | 57 |
| 5.22.2.1 | BMLSpeech() | 57 |
| 5.22.3 | Member Function Documentation | 57 |
| 5.22.3.1 | Parse() | 57 |
| 5.22.4 | Member Data Documentation | 58 |
| 5.22.4.1 | text | 58 |

| | |
|---|----|
| 5.23 BMLNet.BMLStance Class Reference | 58 |
| 5.23.1 Detailed Description | 59 |
| 5.23.2 Member Enumeration Documentation | 59 |
| 5.23.2.1 Type | 59 |
| 5.23.3 Constructor & Destructor Documentation | 59 |
| 5.23.3.1 BMLStance() | 59 |
| 5.23.4 Member Function Documentation | 59 |
| 5.23.4.1 Parse() | 59 |
| 5.23.5 Member Data Documentation | 60 |
| 5.23.5.1 type | 60 |
| 5.24 BMLNet.BMLSyncPoint Class Reference | 60 |
| 5.24.1 Detailed Description | 60 |
| 5.24.2 Constructor & Destructor Documentation | 60 |
| 5.24.2.1 BMLSyncPoint() | 60 |
| 5.24.3 Member Function Documentation | 61 |
| 5.24.3.1 IsCompleted() | 61 |
| 5.24.3.2 IsTimerSafe() | 61 |
| 5.24.3.3 TriggerSyncPoint() | 61 |
| 5.24.3.4 Update() | 62 |
| 5.25 BMLNet.BMLWait Class Reference | 62 |
| 5.25.1 Constructor & Destructor Documentation | 62 |
| 5.25.1.1 BMLWait() | 62 |
| 5.25.2 Member Function Documentation | 63 |
| 5.25.2.1 Parse() | 63 |
| 5.25.3 Member Data Documentation | 63 |
| 5.25.3.1 duration | 63 |
| 5.26 AssetManagerPackage.Dependencies Class Reference | 63 |
| 5.26.1 Detailed Description | 64 |
| 5.26.2 Constructor & Destructor Documentation | 64 |
| 5.26.2.1 Dependencies() | 64 |

| | |
|--|----|
| 5.27 AssetManagerPackage.Depends Class Reference | 64 |
| 5.27.1 Detailed Description | 64 |
| 5.27.2 Constructor & Destructor Documentation | 64 |
| 5.27.2.1 Depends() | 64 |
| 5.27.3 Property Documentation | 65 |
| 5.27.3.1 maxVersion | 65 |
| 5.27.3.2 minVersion | 65 |
| 5.27.3.3 name | 65 |
| 5.28 AssetPackage.IAsset Interface Reference | 65 |
| 5.28.1 Detailed Description | 66 |
| 5.28.2 Property Documentation | 66 |
| 5.28.2.1 Bridge | 66 |
| 5.28.2.2 Class | 66 |
| 5.28.2.3 Dependencies | 66 |
| 5.28.2.4 Id | 67 |
| 5.28.2.5 Maturity | 67 |
| 5.28.2.6 Settings | 67 |
| 5.28.2.7 Version | 67 |
| 5.29 AssetPackage.IBridge Interface Reference | 67 |
| 5.29.1 Detailed Description | 67 |
| 5.30 AssetPackage.IDataStorage Interface Reference | 68 |
| 5.30.1 Detailed Description | 68 |
| 5.30.2 Member Function Documentation | 68 |
| 5.30.2.1 Delete() | 68 |
| 5.30.2.2 Exists() | 68 |
| 5.30.2.3 Files() | 69 |
| 5.30.2.4 Load() | 69 |
| 5.30.2.5 Save() | 69 |
| 5.31 AssetPackage.IDefaultSettings Interface Reference | 70 |
| 5.31.1 Detailed Description | 70 |

| | | |
|----------|---|----|
| 5.31.2 | Member Function Documentation | 71 |
| 5.31.2.1 | HasDefaultSettings() | 71 |
| 5.31.2.2 | LoadDefaultSettings() | 72 |
| 5.31.2.3 | SaveDefaultSettings() | 72 |
| 5.32 | AssetPackage.ILog Interface Reference | 73 |
| 5.32.1 | Detailed Description | 73 |
| 5.32.2 | Member Function Documentation | 73 |
| 5.32.2.1 | Log() | 73 |
| 5.33 | AssetPackage.ISettings Interface Reference | 73 |
| 5.33.1 | Detailed Description | 74 |
| 5.34 | AssetPackage.IWebServiceRequest Interface Reference | 74 |
| 5.34.1 | Detailed Description | 74 |
| 5.34.2 | Member Function Documentation | 74 |
| 5.34.2.1 | WebServiceRequest() | 74 |
| 5.35 | AssetPackage.IWebServiceRequestAsync Interface Reference | 75 |
| 5.35.1 | Detailed Description | 75 |
| 5.35.2 | Member Function Documentation | 75 |
| 5.35.2.1 | WebServiceRequestAsync() | 75 |
| 5.36 | AssetPackage.IWebServiceResponseAsync Interface Reference | 76 |
| 5.36.1 | Detailed Description | 76 |
| 5.36.2 | Member Function Documentation | 76 |
| 5.36.2.1 | Error() | 76 |
| 5.36.2.2 | Success() | 76 |
| 5.37 | AssetPackage.RageBMLNet Class Reference | 77 |
| 5.37.1 | Detailed Description | 78 |
| 5.37.2 | Constructor & Destructor Documentation | 78 |
| 5.37.2.1 | RageBMLNet() | 78 |
| 5.37.3 | Member Function Documentation | 78 |
| 5.37.3.1 | GetBehaviorFromId() | 78 |
| 5.37.3.2 | SyncPointCompleted() | 78 |

| | | |
|----------|---|----|
| 5.37.3.3 | TriggerSyncPoint() | 79 |
| 5.37.3.4 | Update() | 79 |
| 5.37.4 | Property Documentation | 79 |
| 5.37.4.1 | ScheduledBlocks | 79 |
| 5.37.4.2 | Settings | 80 |
| 5.37.4.3 | Timer | 80 |
| 5.38 | AssetPackage.RageBMLNetSettings Class Reference | 80 |
| 5.38.1 | Detailed Description | 81 |
| 5.38.2 | Constructor & Destructor Documentation | 81 |
| 5.38.2.1 | RageBMLNetSettings() | 81 |
| 5.39 | AssetManagerPackage.RageVersionInfo Class Reference | 81 |
| 5.39.1 | Detailed Description | 82 |
| 5.39.2 | Constructor & Destructor Documentation | 82 |
| 5.39.2.1 | RageVersionInfo() | 82 |
| 5.39.3 | Member Function Documentation | 82 |
| 5.39.3.1 | LoadVersionInfo() | 82 |
| 5.39.3.2 | SaveVersionInfo() | 82 |
| 5.39.4 | Property Documentation | 82 |
| 5.39.4.1 | Build | 82 |
| 5.39.4.2 | Dependencies | 83 |
| 5.39.4.3 | Id | 83 |
| 5.39.4.4 | Major | 83 |
| 5.39.4.5 | Maturity | 83 |
| 5.39.4.6 | Minor | 83 |
| 5.39.4.7 | Revision | 83 |
| 5.40 | AssetPackage.RequestResponse Class Reference | 84 |
| 5.40.1 | Detailed Description | 84 |
| 5.40.2 | Constructor & Destructor Documentation | 84 |
| 5.40.2.1 | RequestResponse() [1/2] | 84 |
| 5.40.2.2 | RequestResponse() [2/2] | 84 |

| | | |
|--------------|--|-----------|
| 5.40.3 | Member Data Documentation | 85 |
| 5.40.3.1 | responseCode | 85 |
| 5.40.3.2 | responseHeaders | 85 |
| 5.40.3.3 | responsMessage | 85 |
| 5.40.4 | Property Documentation | 85 |
| 5.40.4.1 | ResultAllowed | 85 |
| 5.41 | AssetPackage.RequestSetttings Class Reference | 85 |
| 5.41.1 | Detailed Description | 86 |
| 5.41.2 | Constructor & Destructor Documentation | 86 |
| 5.41.2.1 | RequestSetttings() | 86 |
| 5.41.3 | Member Data Documentation | 86 |
| 5.41.3.1 | allowedResponsCodes | 86 |
| 5.41.3.2 | body | 86 |
| 5.41.3.3 | method | 87 |
| 5.41.3.4 | requestHeaders | 87 |
| 5.41.3.5 | uri | 87 |
| 5.42 | AssetManagerPackage.StringWriterUtf8 Class Reference | 87 |
| 5.42.1 | Detailed Description | 87 |
| 5.42.2 | Property Documentation | 87 |
| 5.42.2.1 | Encoding | 87 |
| Index | | 89 |

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

| | |
|--|----|
| AssetManagerPackage | |
| Embedded Resources & Android (does seem to apply to pure android code, not the pcl/sap): | 9 |
| AssetPackage | 10 |
| BMLNet | 11 |

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

| | |
|--|----|
| AssetManagerPackage.AssetManager | 13 |
| BMLNet.BMLBlock | 26 |
| BMLNet.BMLBehavior | 24 |
| BMLNet.BMLFace | 32 |
| BMLNet.BMLFaceFacs | 34 |
| BMLNet.BMLFaceLexeme | 36 |
| BMLNet.BMLFaceShift | 37 |
| BMLNet.BMLGaze | 38 |
| BMLNet.BMLGazeShift | 41 |
| BMLNet.BMLGesture | 42 |
| BMLNet.BMLHead | 44 |
| BMLNet.BMLHeadDirectionShift | 46 |
| BMLNet.BMLLocomotion | 48 |
| BMLNet.BMLPointing | 50 |
| BMLNet.BMLPose | 52 |
| BMLNet.BMLPosture | 54 |
| BMLNet.BMLPostureShift | 55 |
| BMLNet.BMLSpeech | 56 |
| BMLNet.BMLStance | 58 |
| BMLNet.BMLWait | 62 |
| BMLNet.BMLBml | 30 |
| BMLNet.BMLFeedback | 38 |
| BMLNet.BMLSyncPoint | 60 |
| AssetManagerPackage.Depends | 64 |
| AssetPackage.IAsset | 65 |
| AssetPackage.BaseAsset | 17 |
| AssetPackage.RageBMLNet | 77 |
| AssetPackage.IBridge | 67 |
| AssetPackage.IDataStorage | 68 |
| AssetPackage.IDefaultSettings | 70 |
| AssetPackage.ILog | 73 |
| AssetPackage.ISettings | 73 |
| AssetPackage.BaseSettings | 23 |
| AssetPackage.RageBMLNetSettings | 80 |

| | |
|---|----|
| AssetPackage.IWebServiceRequest | 74 |
| AssetPackage.IWebServiceRequestAsync | 75 |
| AssetPackage.IWebServiceResponseAsync | 76 |
| List | |
| AssetManagerPackage.Dependencies | 63 |
| AssetManagerPackage.RageVersionInfo | 81 |
| AssetPackage.RequestSettings | 85 |
| AssetPackage.RequestResponse | 84 |
| StringWriter | |
| AssetManagerPackage.StringWriterUtf8 | 87 |

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

| | |
|--|----|
| AssetManagerPackage.AssetManager | 13 |
| Manager for assets. | |
| AssetPackage.BaseAsset | 17 |
| A base asset. | |
| AssetPackage.BaseSettings | 23 |
| A base settings. | |
| BMLNet.BMLBehavior | 24 |
| BML behavior class. all behavior need to derived from this class | |
| BMLNet.BMLBlock | 26 |
| abstract class of BML block all block need to be derived from this class | |
| BMLNet.BMLBml | 30 |
| BMLNet.BMLFace | 32 |
| Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit. | |
| BMLNet.BMLFaceFacs | 34 |
| This behavior provides control of the face through single Action Units from the Facial Action Coding Scheme. It is an Core Extension, that is, not every BML Compliant Realizer has to implement this behavior, but if a Realizer offers FACS based face control, they should adhere to the specification of this <faceFacs> behavior | |
| BMLNet.BMLFaceLexeme | 36 |
| Show a (partial) face expression from a predefined lexicon. | |
| BMLNet.BMLFaceShift | 37 |
| Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit, where the specified compound face expression becomes the new BASE state of the ECAs face. | |
| BMLNet.BMLFeedback | 38 |
| BMLNet.BMLGaze | 38 |
| Temporarily directs the gaze of the character towards a target. This behavior causes the character to temporarily direct its gaze to the requested target. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera. | |
| BMLNet.BMLGazeShift | 41 |
| Permanently change the gaze direction of the character towards a certain target. This behavior causes the character to direct its gaze to the requested target. This changes the default state of the ECA: after completing this behavior, the new target is the default gaze direction of the ECA. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera. | |

| | |
|--|----|
| BMLNet.BMLGesture | |
| Currently, BML offers two types of gesture behaviors. The first provides a set of gestures recalled by name from a gesticon; the second provides simple pointing gestures. Coordinated movement with arms and hands, recalled from a gesticon by requesting the corresponding lexeme | 42 |
| BMLNet.BMLHead | |
| Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. The stroke phase of the head motion (from strokeStart till strokeEnd is the "meaningful" part of the head motion. The stroke sync point is the "peak" moment of the motion. If repetition > 1, the meaning of the stroke sync point becomes undefined | 44 |
| BMLNet.BMLHeadDirectionShift | |
| Orient the head towards a target referenced by the target attribute. Permanently orient the head in a certain direction. | 46 |
| BMLNet.BMLLocomotion | |
| Move the body of the character from one location to another. This behavior causes the character to move to the requested target in the manner described. | 48 |
| BMLNet.BMLPointing | |
| Deictic gesture towards the target specified by the target attribute | 50 |
| BMLNet.BMLPose | |
| Child element of <posture> and <postureShift> behaviors, defines additions to the global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines additions that modify the global body posture of the ECA. For each value of the part attribute, only one <pose> child is expected to be present. A BML Realizer may define any number of lexemes beyond the ones specified above. | 52 |
| BMLNet.BMLPosture | |
| Temporarily change the posture of the ECA. Temporarily change the posture of the ECA. After the <posture> behavior has ended, return to the BASE posture. | 54 |
| BMLNet.BMLPostureShift | |
| Permanently change the BASE posture of the ECA. | 55 |
| BMLNet.BMLSpeech | |
| Utterance to be spoken by a character. Realization of the <speech> element generates both speech audio (or text) and speech movement, for example using a speech synthesizer and viseme morphing. The <speech> element requires a sub-element. This sub-element is a <text> element that contains the text to be spoken, with optionally embedded <sync> elements for alignment with other behaviors. | 56 |
| BMLNet.BMLStance | |
| Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. This global posture may then be modified by one or more <pose> siblings. | 58 |
| BMLNet.BMLSyncPoint | |
| BML Sync Point class possible format: behavior_id:sync_id [+/- offset] A reference to a sync point of another behavior, optionally with a float offset in seconds. By default, this is a behavior in the same <bml> block that the syncref is contained in; if optional prefix block_id: is present, the syncref specifies a sync point of a behavior in the <bml> block with that ID.) offset: A positive float offset in seconds relative to the start time of the surrounding <bml> block | 60 |
| BMLNet.BMLWait | |
| | 62 |
| AssetManagerPackage.Dependencies | |
| A dependencies. | 63 |
| AssetManagerPackage.Depends | |
| A dependency. | 64 |
| AssetPackage.IAsset | |
| Interface for asset. | 65 |
| AssetPackage.IBridge | |
| Interface for bridge. | 67 |
| AssetPackage.IDataStorage | |
| Interface for data storage. | 68 |

[AssetPackage.IDefaultSettings](#)

Interface for default settings.

This Interface is used to:

- Check if an asset has default (application) settings that override build-in default settings.
- Load these settings from the game environment.
- In certain environments write the actual settings as application defaults. This could for instance be Unity in editor mode.

70

[AssetPackage.ILog](#)

Interface for logger. 73

[AssetPackage.ISettings](#)

Interface for settings. 73

[AssetPackage.IWebServiceRequest](#)

Interface for web service request. 74

[AssetPackage.IWebServiceRequestAsync](#)

Interface for web service request. 75

[AssetPackage.IWebServiceResponseAsync](#)

Interface for web service response. 76

[AssetPackage.RageBMLNet](#)

An BMLNet Rage asset 77

[AssetPackage.RageBMLNetSettings](#)

An asset settings 80

[AssetManagerPackage.RageVersionInfo](#)

Information about the rage version. 81

[AssetPackage.RequestResponse](#)

Response results. 84

[AssetPackage.RequestSettings](#)

Request Settings. 85

[AssetManagerPackage.StringWriterUtf8](#)

A StringWriter UTF8. 87

Chapter 4

Namespace Documentation

4.1 AssetManagerPackage Namespace Reference

Embedded Resources & Android (does seem to apply to pure android code, not the pcl/sap):

Classes

- class [AssetManager](#)
Manager for assets.
- class [Dependencies](#)
A dependencies.
- class [Depends](#)
A dependency.
- class **Messages**
A Broadcast Messages class.
- class **RageExtensions**
A rage extensions.
- class [RageVersionInfo](#)
Information about the rage version.
- class [StringWriterUtf8](#)
A StringWriter UTF8.

4.1.1 Detailed Description

Embedded Resources & Android (does seem to apply to pure android code, not the pcl/sap):

Naming: Deployment Localization: WinPhone/PCL

4.2 AssetPackage Namespace Reference

Classes

- class [BaseAsset](#)
A base asset.
- class [BaseSettings](#)
A base settings.
- interface [IAsset](#)
Interface for asset.
- interface [IBridge](#)
Interface for bridge.
- interface [IDataStorage](#)
Interface for data storage.
- interface [IDefaultSettings](#)
Interface for default settings.

This Interface is used to:
 - Check if an asset has default (application) settings that override build-in default settings.
 - Load these settings from the game environment.
 - In certain environments write the actual settings as application defaults. This could for instance be Unity in editor mode.
- interface [ILog](#)
Interface for logger.
- interface [ISettings](#)
Interface for settings.
- interface [IWebServiceRequest](#)
Interface for web service request.
- interface [IWebServiceRequestAsync](#)
Interface for web service request.
- interface [IWebServiceResponseAsync](#)
Interface for web service response.
- class [RageBMLNet](#)
An *BMLNet* Rage asset
- class [RageBMLNetSettings](#)
An asset settings.
- class [RequestResponse](#)
Response results.
- class [RequestSettings](#)
Request Settings.

Enumerations

- enum [Severity](#) : int {
[Severity.Critical](#) = 1, [Severity.Error](#) = 2, [Severity.Warning](#) = 4, [Severity.Information](#) = 8,
[Severity.Verbose](#) = 16 }
 Values that represent log severity.
 See [Trace and Event Log Severity Levels](#)
- enum [LogLevel](#) : int {
[LogLevel.Critical](#) = [Severity.Critical](#), [LogLevel.Error](#) = [Critical](#) | [Severity.Error](#), [LogLevel.Warn](#) = [Error](#) | [Severity.Warning](#), [LogLevel.Info](#) = [Warn](#) | [Severity.Information](#),
[LogLevel.All](#) = [Severity.Critical](#) | [Severity.Error](#) | [Severity.Warning](#) | [Severity.Information](#) | [Severity.Verbose](#) }
 Values that represent log levels.

4.2.1 Enumeration Type Documentation

4.2.1.1 LogLevel

```
enum AssetPackage.LogLevel : int [strong]
```

Values that represent log levels.

Enumerator

| | |
|----------|---|
| Critical | An enum constant representing the critical option. |
| Error | An enum constant representing the error option. |
| Warn | An enum constant representing the warning option. |
| Info | An enum constant representing the information option. |
| All | An enum constant representing all option. |

4.2.1.2 Severity

```
enum AssetPackage.Severity : int [strong]
```

Values that represent log severity.

See [Trace and Event Log Severity Levels](#)

Enumerator

| | |
|-------------|---|
| Critical | An enum constant representing the critical option. |
| Error | An enum constant representing the error option. |
| Warning | An enum constant representing the warning option. |
| Information | An enum constant representing the information option. |
| Verbose | An enum constant representing the verbose option. |

4.3 BMLNet Namespace Reference

Classes

- class [BMLBehavior](#)
BML behavior class. all behavior need to derived from this class
- class [BMLBlock](#)
abstract class of BML block all block need to be derived from this class
- class [BMLBml](#)
- class [BMLFace](#)
Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit.
- class [BMLFaceFacs](#)
This behavior provides control of the face through single Action Units from the Facial Action Coding Scheme. It is an Core Extension, that is, not every BML Compliant Realizer has to implement this behavior, but if a Realizer offers FACS based face control, they should adhere to the specification of this <faceFacs> behavior

- class [BMLFaceLexeme](#)
Show a (partial) face expression from a predefined lexicon.
- class [BMLFaceShift](#)
Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit, where the specified compound face expression becomes the new BASE state of the ECAs face.
- class [BMLFeedback](#)
- class [BMLGaze](#)
Temporarily directs the gaze of the character towards a target. This behavior causes the character to temporarily direct its gaze to the requested target. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.
- class [BMLGazeShift](#)
Permanently change the gaze direction of the character towards a certain target. This behavior causes the character to direct its gaze to the requested target. This changes the default state of the ECA: after completing this behavior, the new target is the default gaze direction of the ECA. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.
- class [BMLGesture](#)
Currently, BML offers two types of gesture behaviors. The first provides a set of gestures recalled by name from a gesticon; the second provides simple pointing gestures. Coordinated movement with arms and hands, recalled from a gesticon by requesting the corresponding lexeme
- class [BMLHead](#)
Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. The stroke phase of the head motion (from strokeStart till strokeEnd is the "meaningful" part of the head motion. The stroke sync point is the "peak" moment of the motion. If repetition > 1, the meaning of the stroke sync point becomes undefined
- class [BMLHeadDirectionShift](#)
Orient the head towards a target referenced by the target attribute. Permanently orient the head in a certain direction.
- class [BMLLocomotion](#)
Move the body of the character from one location to another. This behavior causes the character to move to the requested target in the manner described.
- class [BMLPointing](#)
Deictic gesture towards the target specified by the target attribute
- class [BMLPose](#)
Child element of <posture> and <postureShift> behaviors, defines additions to the global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines additions that modify the global body posture of the ECA. For each value of the part attribute, only one <pose> child is expected to be present. A BML Realizer may define any number of lexemes beyond the ones specified above.
- class [BMLPosture](#)
Temporarily change the posture of the ECA. Temporarily change the posture of the ECA. After the <posture> behavior has ended, return to the BASE posture.
- class [BMLPostureShift](#)
Permanently change the BASE posture of the ECA.
- class [BMLSpeech](#)
Utterance to be spoken by a character. Realization of the <speech> element generates both speech audio (or text) and speech movement, for example using a speech synthesizer and viseme morphing. The <speech> element requires a sub-element. This sub-element is a <text> element that contains the text to be spoken, with optionally embedded <sync> elements for alignment with other behaviors.
- class [BMLStance](#)
Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. This global posture may then be modified by one or more <pose> siblings.
- class [BMLSyncPoint](#)
BML Sync Point class possible format: behavior_id:sync_id [+/- offset] A reference to a sync point of another behavior, optionally with a float offset in seconds. By default, this is a behavior in the same <bml> block that the syncref is contained in; if optional prefix block_id: is present, the syncref specifies a sync point of a behavior in the <bml> block with that ID.) offset: A positive float offset in seconds relative to the start time of the surrounding <bml> block.
- class [BMLWait](#)

Chapter 5

Class Documentation

5.1 AssetManagerPackage.AssetManager Class Reference

Manager for assets.

Public Member Functions

- [IAsset findAssetByClass](#) (String claz)
Searches for the first asset by class.
- [IAsset findAssetById](#) (String id)
Searches for the first asset by identifier.
- List< [IAsset](#) > [findAssetsByClass](#) (String claz)
Searches for assets by class.
- String [registerAssetInstance](#) ([IAsset](#) asset, String claz)
Registers the asset instance.
- void [Log](#) ([Severity](#) loglevel, String format, params object[] args)
Logs.
- void [Log](#) ([Severity](#) loglevel, String msg)
Logs.

Protected Member Functions

- T [getInterface](#)< T > ()
Gets the interface.

Properties

- static [AssetManager Instance](#) [get]
Visible when reflecting.
- [IBridge Bridge](#) [get, set]
Gets or sets the bridge.
- String [VersionAndDependenciesReport](#) [get]
Reports version and dependencies.

5.1.1 Detailed Description

Manager for assets.

5.1.2 Member Function Documentation

5.1.2.1 findAssetByClass()

```
IAsset AssetManagerPackage.AssetManager.findAssetByClass (
    String claz )
```

Searches for the first asset by class.

Parameters

| | |
|-------------|-----------|
| <i>claz</i> | The claz. |
|-------------|-----------|

Returns

The found asset by class.

5.1.2.2 findAssetById()

```
IAsset AssetManagerPackage.AssetManager.findAssetById (
    String id )
```

Searches for the first asset by identifier.

Parameters

| | |
|-----------|-----------------|
| <i>id</i> | The identifier. |
|-----------|-----------------|

Returns

The found asset by identifier.

5.1.2.3 findAssetsByClass()

```
List<IAsset> AssetManagerPackage.AssetManager.findAssetsByClass (
    String claz )
```

Searches for assets by class.

Parameters

| | |
|-------------|-----------|
| <i>claz</i> | The claz. |
|-------------|-----------|

Returns

The found assets by class.

5.1.2.4 `getInterface< T >()`

```
T AssetManagerPackage.AssetManager.getInterface< T > ( ) [protected]
```

Gets the interface.

Template Parameters

| | |
|----------|-------------------------|
| <i>T</i> | Generic type parameter. |
|----------|-------------------------|

Returns

The interface.

5.1.2.5 `Log()` [1/2]

```
void AssetManagerPackage.AssetManager.Log (
    Severity loglevel,
    String format,
    params object [] args )
```

Logs.

Parameters

| | |
|-----------------|---|
| <i>loglevel</i> | The loglevel. |
| <i>format</i> | Describes the format to use. |
| <i>args</i> | A variable-length parameters list containing arguments. |

5.1.2.6 `Log()` [2/2]

```
void AssetManagerPackage.AssetManager.Log (
    Severity loglevel,
    String msg )
```

Logs.

Parameters

| | |
|-----------------|---------------|
| <i>loglevel</i> | The loglevel. |
| <i>msg</i> | The message. |

5.1.2.7 registerAssetInstance()

```
String AssetManagerPackage.AssetManager.registerAssetInstance (
    IAsset asset,
    String claz )
```

Registers the asset instance.

Parameters

| | |
|--------------|------------|
| <i>asset</i> | The asset. |
| <i>claz</i> | The claz. |

Returns

A String.

5.1.3 Property Documentation

5.1.3.1 Bridge

```
IBridge AssetManagerPackage.AssetManager.Bridge [get], [set]
```

Gets or sets the bridge.

The bridge.

5.1.3.2 Instance

```
AssetManager AssetManagerPackage.AssetManager.Instance [static], [get]
```

Visible when reflecting.

The instance.

5.1.3.3 VersionAndDependenciesReport

```
String AssetManagerPackage.AssetManager.VersionAndDependenciesReport [get]
```

Reports version and dependencies.

The version and dependencies report.

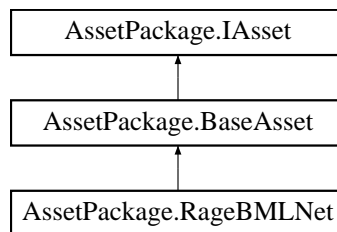
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/AssetManager.cs

5.2 AssetPackage.BaseAsset Class Reference

A base asset.

Inheritance diagram for AssetPackage.BaseAsset:



Public Member Functions

- [BaseAsset](#) ()
Initializes a new instance of the AssetManagerPackage.BaseAsset class.
- [BaseAsset](#) (IBridge bridge)
Initializes a new instance of the AssetPackage.BaseAsset class.
- void [Log](#) (Severity severity, String format, params object[] args)
Logs.
- void [Log](#) (Severity severity, String msg)
Logs.
- Boolean [LoadDefaultSettings](#) ()
Loads Settings object from Default (Design-time) Settings.
- Boolean [LoadSettings](#) (String filename)
Loads Settings object as Run-time Settings.
- Boolean [SaveDefaultSettings](#) (bool force)
Saves Settings object as Default (Design-time) Settings.
- Boolean [SaveSettings](#) (String filename)
Save Settings object from Run-time Settings.
- [ISettings SettingsFromXml](#) (String xml)
Settings from XML.
- String [SettingsToXml](#) ()
Settings to XML.

Protected Member Functions

- String [GetEmbeddedResource](#) (String ns, String res)
Gets embedded resource.
- T [getInterface< T >](#) ()
Gets the interface.

Properties

- [IBridge Bridge](#) [get, set]
Gets or sets the bridge.
- String [Class](#) [get]
Gets the class.
- Dictionary< String, String > [Dependencies](#) [get]
Gets the dependencies.
- Boolean [hasSettings](#) [get]
Gets a value indicating whether this object has settings.
- String [Id](#) [get]
Gets the identifier.
- String [Maturity](#) [get]
Gets the maturity.
- virtual [ISettings Settings](#) [get, set]
Gets or sets options for controlling the operation.
- String [Version](#) [get]
Gets the version.
- [RageVersionInfo VersionInfo](#) [get]
Gets information describing the version.

5.2.1 Detailed Description

A base asset.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 BaseAsset() [1/2]

```
AssetPackage.BaseAsset.BaseAsset ( )
```

Initializes a new instance of the `AssetManagerPackage.BaseAsset` class.

List Embedded Resources.

5.2.2.2 BaseAsset() [2/2]

```
AssetPackage.BaseAsset.BaseAsset (
    IBridge bridge )
```

Initializes a new instance of the `AssetPackage.BaseAsset` class.

Parameters

| | |
|---------------|-------------|
| <i>bridge</i> | The bridge. |
|---------------|-------------|

5.2.3 Member Function Documentation

5.2.3.1 GetEmbeddedResource()

```
String AssetPackage.BaseAsset.GetEmbeddedResource (
    String ns,
    String res ) [protected]
```

Gets embedded resource.

Parameters

| | |
|------------|--------------------|
| <i>ns</i> | The namespace. |
| <i>res</i> | The resource name. |

Returns

The embedded resource.

0) AppDomain is not present in Unity/WP81

0) Returns RageAssetManager.dll instead of the asset

1) Compiles but fails on Unity/WP81 with a console dump

5.2.3.2 getInterface< T >()

```
T AssetPackage.BaseAsset.getInterface< T > ( ) [protected]
```

Gets the interface.

Template Parameters

| | |
|----------|-------------------------|
| <i>T</i> | Generic type parameter. |
|----------|-------------------------|

Returns

The interface.

5.2.3.3 LoadDefaultSettings()

```
Boolean AssetPackage.BaseAsset.LoadDefaultSettings ( )
```

Loads Settings object from Default (Design-time) Settings.

In Unity Resources.Load() must be used and the files will be loaded a Assets\Resources Folder.

Returns

true if it succeeds, false if it fails.

5.2.3.4 LoadSettings()

```
Boolean AssetPackage.BaseAsset.LoadSettings (
    String filename )
```

Loads Settings object as Run-time Settings.

The resulting file will be read using the [IDataStorage](#) interface.

Parameters

| | |
|-----------------|-----------------------|
| <i>filename</i> | Filename of the file. |
|-----------------|-----------------------|

Returns

true if it succeeds, false if it fails.

5.2.3.5 Log() [1/2]

```
void AssetPackage.BaseAsset.Log (
    Severity severity,
    String format,
    params object [] args )
```

Logs.

Parameters

| | |
|-----------------|---|
| <i>severity</i> | The severity. |
| <i>format</i> | Describes the format to use. |
| <i>args</i> | A variable-length parameters list containing arguments. |

5.2.3.6 Log() [2/2]

```
void AssetPackage.BaseAsset.Log (
    Severity severity,
    String msg )
```

Logs.

Parameters

| | |
|-----------------|---------------|
| <i>severity</i> | The severity. |
| <i>msg</i> | The message. |

5.2.3.7 SaveDefaultSettings()

```
Boolean AssetPackage.BaseAsset.SaveDefaultSettings (
```

```
bool force )
```

Saves Settings object as Default (Design-time) Settings.

In Unity the file will be saved in a Assets\Resources Folder in the editor environment (As resources are read-only at run-time).

Returns

true if it succeeds, false if it fails.

5.2.3.8 SaveSettings()

```
Boolean AssetPackage.BaseAsset.SaveSettings (
    String filename )
```

Save Settings object from Run-time Settings.

The resulting file will be written using the [IDataStorage](#) interface.

Parameters

| | |
|-----------------|-----------------------|
| <i>filename</i> | Filename of the file. |
|-----------------|-----------------------|

Returns

true if it succeeds, false if it fails.

5.2.3.9 SettingsFromXml()

```
ISettings AssetPackage.BaseAsset.SettingsFromXml (
    String xml )
```

Settings from XML.

Parameters

| | |
|------------|----------|
| <i>xml</i> | The XML. |
|------------|----------|

Returns

The [ISettings](#).

Use DataContractSerializer or DataContractJsonSerializer?

5.2.3.10 SettingsToXml()

```
String AssetPackage.BaseAsset.SettingsToXml ( )
```

Settings to XML.

Returns

A String.

Use `DataContractSerializer` or `DataContractJsonSerializer`?

5.2.4 Property Documentation

5.2.4.1 Bridge

```
IBridge AssetPackage.BaseAsset.Bridge [get], [set]
```

Gets or sets the bridge.

The bridge.

5.2.4.2 Class

```
String AssetPackage.BaseAsset.Class [get]
```

Gets the class.

The class.

5.2.4.3 Dependencies

```
Dictionary<String, String> AssetPackage.BaseAsset.Dependencies [get]
```

Gets the dependencies.

The dependencies.

5.2.4.4 hasSettings

```
Boolean AssetPackage.BaseAsset.hasSettings [get]
```

Gets a value indicating whether this object has settings.

true if this object has settings, false if not.

5.2.4.5 Id

```
String AssetPackage.BaseAsset.Id [get]
```

Gets the identifier.

The identifier.

5.2.4.6 Maturity

```
String AssetPackage.BaseAsset.Maturity [get]
```

Gets the maturity.

The maturity.

5.2.4.7 Settings

```
virtual ISettings AssetPackage.BaseAsset.Settings [get], [set]
```

Gets or sets options for controlling the operation.

The settings.

5.2.4.8 Version

```
String AssetPackage.BaseAsset.Version [get]
```

Gets the version.

The version.

5.2.4.9 VersionInfo

```
RageVersionInfo AssetPackage.BaseAsset.VersionInfo [get]
```

Gets information describing the version.

Information describing the version.

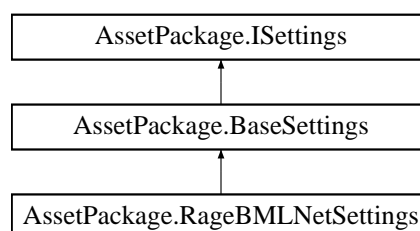
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/BaseAsset.cs

5.3 AssetPackage.BaseSettings Class Reference

A base settings.

Inheritance diagram for AssetPackage.BaseSettings:



Public Member Functions

- [BaseSettings](#) ()
Initializes a new instance of the Swiss.BaseSettings class.

Static Public Member Functions

- static void [UpdateDefaultValues](#) (Object obj)
Set the value of (Public Instance) properties to the DefaultValueAttribute's Value of that property.

5.3.1 Detailed Description

A base settings.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 BaseSettings()

```
AssetPackage.BaseSettings.BaseSettings ( )
```

Initializes a new instance of the Swiss.BaseSettings class.

Initialize Settings to their specified default values.

5.3.3 Member Function Documentation

5.3.3.1 UpdateDefaultValues()

```
static void AssetPackage.BaseSettings.UpdateDefaultValues (
    Object obj ) [static]
```

Set the value of (Public Instance) properties to the DefaultValueAttribute's Value of that property.

Parameters

| | |
|------------|-------------|
| <i>obj</i> | The object. |
|------------|-------------|

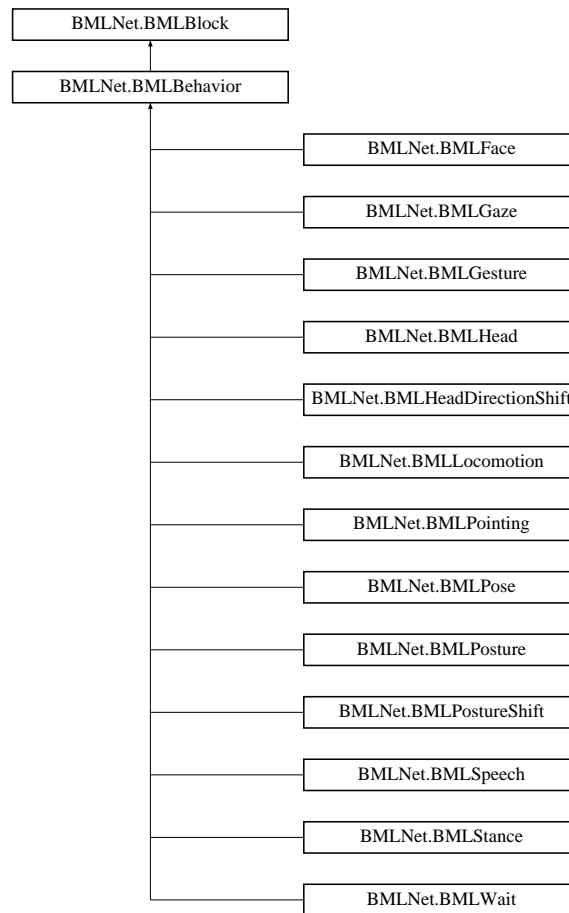
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/BaseSettings.cs↔

5.4 BMLNet.BMLBehavior Class Reference

BML behavior class. all behavior need to derived from this class

Inheritance diagram for BMLNet.BMLBehavior:



Public Member Functions

- [BMLBehavior](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id

Additional Inherited Members

5.4.1 Detailed Description

BML behavior class. all behavior need to derived from this class

5.4.2 Constructor & Destructor Documentation

5.4.2.1 BMLBehavior()

```
BMLNet.BMLBehavior.BMLBehavior ( )
```

constructor

5.4.3 Member Function Documentation

5.4.3.1 Parse()

```
override void BMLNet.BMLBehavior.Parse (  
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Implements [BMLNet.BMLBlock](#).

Reimplemented in [BMLNet.BMLGaze](#), [BMLNet.BMLPose](#), [BMLNet.BMLFaceLexeme](#), [BMLNet.BMLGesture](#), [BMLNet.BMLHead](#), [BMLNet.BMLPointing](#), [BMLNet.BMLFace](#), [BMLNet.BMLLocomotion](#), [BMLNet.BMLStance](#), [BMLNet.BMLSpeech](#), [BMLNet.BMLHeadDirectionShift](#), [BMLNet.BMLGazeShift](#), [BMLNet.BMLPosture](#), [BMLNet.BMLWait](#), and [BMLNet.BMLPostureShift](#).

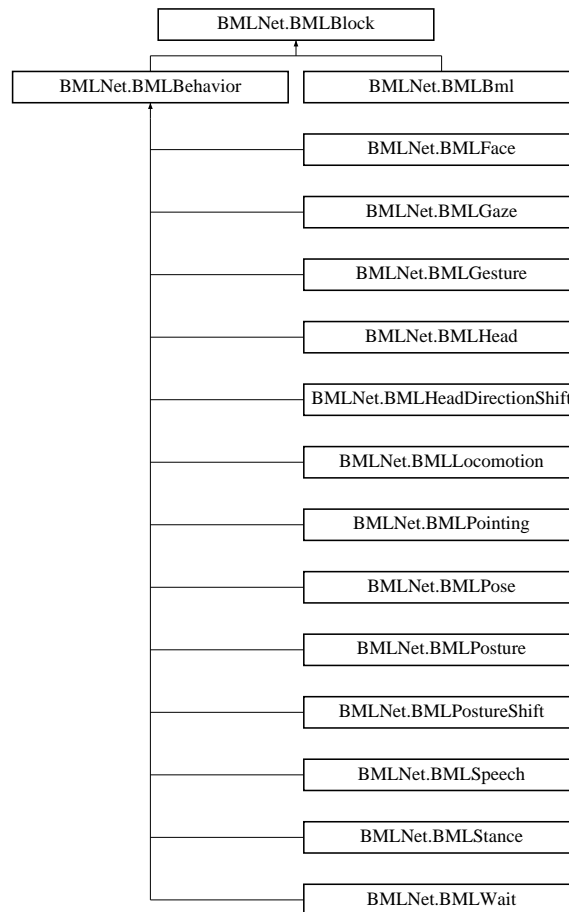
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLBehavior.cs

5.5 BMLNet.BMLBlock Class Reference

abstract class of BML block all block need to be derived from this class

Inheritance diagram for BMLNet.BMLBlock:



Public Member Functions

- [BMLBlock](#) ()
empty constructor
- abstract void [Parse](#) (XmlReader reader)
all child class need to implement their own parsing standard
- void **Dispose** ()
- String **getCharacterId** ()

Public Attributes

- string [id](#)
Unique ID that allows referencing to a particular bml block. The id 'bml' is reserved.
- Dictionary< string, [BMLSyncPoint](#) > [syncPoints](#) = new Dictionary<string, [BMLSyncPoint](#)>()
Sync Point collection of this block
- [BMLBml](#) [parentBml](#)
parent bml tag

Protected Member Functions

- T [TryParseAttribute](#)< T > (XmlReader reader, string attributeName, T defaultValue, bool required=true)
helper function to parse the attribute from XML
- bool [TryParseSyncPoint](#) (XmlReader reader, string eventName)
helper function to parse the sync point attribute we do not need to check whether we found the attribute or not. The [BMLSyncPoint](#) class will use those value (null or not null).

5.5.1 Detailed Description

abstract class of BML block all block need to be derived from this class

5.5.2 Constructor & Destructor Documentation

5.5.2.1 BMLBlock()

```
BMLNet.BMLBlock.BMLBlock ( )
```

empty constructor

5.5.3 Member Function Documentation

5.5.3.1 Parse()

```
abstract void BMLNet.BMLBlock.Parse (
    XmlReader reader ) [pure virtual]
```

all child class need to implement their own parsing standard

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Implemented in [BMLNet.BMLGaze](#), [BMLNet.BMLPose](#), [BMLNet.BMLBml](#), [BMLNet.BMLFaceLexeme](#), [BMLNet.BMLGesture](#), [BMLNet.BMLHead](#), [BMLNet.BMLPointing](#), [BMLNet.BMLFace](#), [BMLNet.BMLLocomotion](#), [BMLNet.BMLStance](#), [BMLNet.BMLSpeech](#), [BMLNet.BMLHeadDirectionShift](#), [BMLNet.BMLGazeShift](#), [BMLNet.BMLPosture](#), [BMLNet.BMLWait](#), [BMLNet.BMLBehavior](#), and [BMLNet.BMLPostureShift](#).

5.5.3.2 TryParseAttribute< T >()

```
T BMLNet.BMLBlock.TryParseAttribute< T > (
    XmlReader reader,
    string attributeName,
    T defaultValue,
    bool required = true ) [protected]
```

helper function to parse the attribute from XML

Template Parameters

| | |
|----------|--|
| <i>T</i> | |
|----------|--|

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Parameters

| | |
|----------------------|--|
| <i>attributeName</i> | |
|----------------------|--|

the attribute name that we need to parse

Parameters

| | |
|---------------------|--|
| <i>defaultValue</i> | |
|---------------------|--|

the value when we do not find the attribute

Parameters

| | |
|-----------------|--|
| <i>required</i> | |
|-----------------|--|

do you require this attribute or not ?

Returns

5.5.3.3 TryParseSyncPoint()

```
bool BMLNet.BMLBlock.TryParseSyncPoint (
    XmlReader reader,
    string eventName ) [protected]
```

helper function to parse the sync point attribute we do not need to check whether we found the attribute or not. The [BMLSyncPoint](#) class will use those value (null or not null).

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Parameters

| | |
|------------------|--|
| <i>eventName</i> | |
|------------------|--|

the name of sync point (start, ready, strokeStart, attackPeak, stroke, strokeEnd, relax, end)

Returns

5.5.4 Member Data Documentation

5.5.4.1 id

```
string BMLNet.BMLBlock.id
```

Unique ID that allows referencing to a particular bml block. The id 'bml' is reserved.

5.5.4.2 parentBml

```
BMLBml BMLNet.BMLBlock.parentBml
```

parent bml tag

5.5.4.3 syncPoints

```
Dictionary<string, BMLSyncPoint> BMLNet.BMLBlock.syncPoints = new Dictionary<string, BML↵  
SyncPoint> ()
```

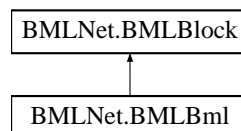
Sync Point collection of this block

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BML↵
Block.cs

5.6 BMLNet.BMLBml Class Reference

Inheritance diagram for BMLNet.BMLBml:



Public Types

- enum [Composition](#) { **MERGE**, **APPEND**, **REPLACE** }

one among [MERGE,APPEND,REPLACE], defines the composition policy to apply if the current <bml> block overlaps with previous <bml> blocks (see below).

Public Member Functions

- [BMLBml](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml
- void **SetGlobalStartTrigger** (string value)
- void **IncreaseChild** ()
- bool **IncreaseEndChild** ()

Public Attributes

- string [characterId](#)
a reference towards the controlled character
- string **xmlns**
- [Composition composition](#)
one among [MERGE,APPEND,REPLACE], defines the composition policy to apply if the current <bml> block overlaps with previous <bml> blocks (see below).

Additional Inherited Members

5.6.1 Member Enumeration Documentation

5.6.1.1 Composition

```
enum BMLNet.BMLBml.Composition [strong]
```

one among [MERGE,APPEND,REPLACE], defines the composition policy to apply if the current <bml> block overlaps with previous <bml> blocks (see below).

5.6.2 Constructor & Destructor Documentation

5.6.2.1 BMLBml()

```
BMLNet.BMLBml.BMLBml ( )
```

constructor

5.6.3 Member Function Documentation

5.6.3.1 Parse()

```
override void BMLNet.BMLBml.Parse (  
    XmlReader reader ) [virtual]
```

parsing the xml

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Implements [BMLNet.BMLBlock](#).

5.6.4 Member Data Documentation

5.6.4.1 characterId

`string BMLNet.BMLBml.characterId`

a reference towards the controlled character

5.6.4.2 composition

[Composition](#) `BMLNet.BMLBml.composition`

one among [MERGE,APPEND,REPLACE], defines the composition policy to apply if the current `<bml>` block overlaps with previous `<bml>` blocks (see below).

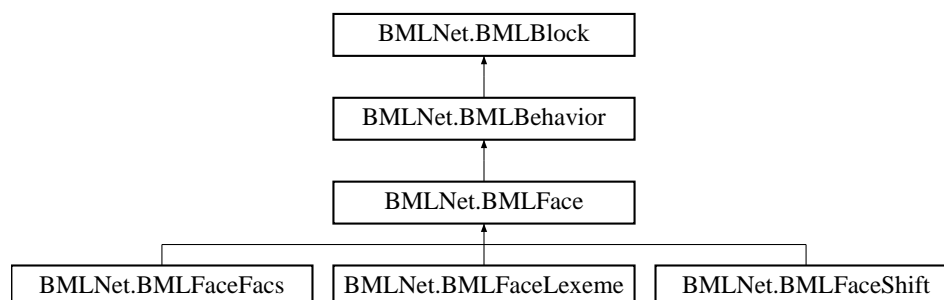
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BML↔Bml.cs

5.7 BMLNet.BMLFace Class Reference

Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit.

Inheritance diagram for BMLNet.BMLFace:



Public Member Functions

- [BMLFace](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id, amount, overshoot synx attribute: start, attackPeak, relax, end

Public Attributes

- float [amount](#)

A float value between 0..1 to indicate the amount to which the expression should be shown on the face, 0 meaning 'not at all' and 1 meaning 'maximum, highly exaggerated'

- float [overshoot](#)

Fraction of overshoot of the attack peak, relative to amount (which defines the level of the sustain phase)

Additional Inherited Members

5.7.1 Detailed Description

Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 BMLFace()

```
BMLNet.BMLFace.BMLFace ( )
```

constructor

5.7.3 Member Function Documentation

5.7.3.1 Parse()

```
override void BMLNet.BMLFace.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id, amount, overshoot synx attribute: start, attackPeak, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Reimplemented from [BMLNet.BMLBehavior](#).

Reimplemented in [BMLNet.BMLFaceLexeme](#).

5.7.4 Member Data Documentation

5.7.4.1 amount

```
float BMLNet.BMLFace.amount
```

A float value between 0..1 to indicate the amount to which the expression should be shown on the face, 0 meaning 'not at all' and 1 meaning 'maximum, highly exaggerated'

5.7.4.2 overshoot

```
float BMLNet.BMLFace.overshoot
```

Fraction of overshoot of the attack peak, relative to amount (which defines the level of the sustain phase)

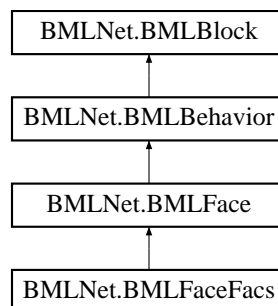
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLFace.cs

5.8 BMLNet.BMLFaceFacs Class Reference

This behavior provides control of the face through single Action Units from the Facial Action Coding Scheme. It is an Core Extension, that is, not every BML Compliant Realizer has to implement this behavior, but if a Realizer offers FACS based face control, they should adhere to the specification of this <faceFacs> behavior

Inheritance diagram for BMLNet.BMLFaceFacs:



Public Types

- enum [Side](#) { **LEFT**, **RIGHT**, **BOTH** }

Public Member Functions

- [BMLFaceFacs](#) ()
Constructor

Public Attributes

- int [au](#)
The number of the FACS Action Unit to be displayed
- [Side](#) [side](#)
Which side of the face to display the action unit on. Possible values: [LEFT,RIGHT,BOTH] Note that for some Action Units, BOTH is the only possible value

Additional Inherited Members

5.8.1 Detailed Description

This behavior provides control of the face through single Action Units from the Facial Action Coding Scheme. It is an Core Extension, that is, not every BML Compliant Realizer has to implement this behavior, but if a Realizer offers FACS based face control, they should adhere to the specification of this <faceFacs> behavior

5.8.2 Member Enumeration Documentation

5.8.2.1 Side

```
enum BMLNet.BMLFaceFacs.Side [strong]
```

5.8.3 Constructor & Destructor Documentation

5.8.3.1 BMLFaceFacs()

```
BMLNet.BMLFaceFacs.BMLFaceFacs ( )
```

Constructor

5.8.4 Member Data Documentation

5.8.4.1 au

```
int BMLNet.BMLFaceFacs.au
```

The number of the FACS Action Unit to be displayed

5.8.4.2 side

```
Side BMLNet.BMLFaceFacs.side
```

Which side of the face to display the action unit on. Possible values: [LEFT,RIGHT,BOTH] Note that for some Action Units, BOTH is the only possible value

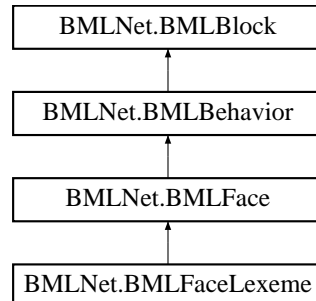
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLFaceFacs.cs

5.9 BMLNet.BMLFaceLexeme Class Reference

Show a (partial) face expression from a predefined lexicon.

Inheritance diagram for BMLNet.BMLFaceLexeme:



Public Types

- enum [Lexeme](#) {
NONE, OBLIQUE_BROWS, RAISE_BROWS, RAISE_LEFT_BROW,
RAISE_RIGHT_BROW, LOWER_BROWS, LOWER_LEFT_BROW, LOWER_RIGHT_BROW,
LOWER_MOUTH_CORNERS, LOWER_LEFT_MOUTH_CORNER, LOWER_RIGHT_MOUTH_CORNER,
RAISE_MOUTH_CORNERS,
RAISE_RIGHT_MOUTH_CORNER, RAISE_LEFT_MOUTH_CORNER, OPEN_MOUTH, OPEN_LIPS,
WIDEN_EYES, CLOSE_EYES }

Public Member Functions

- [BMLFaceLexeme](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: lexeme

Public Attributes

- [Lexeme](#) *lexeme*

Additional Inherited Members

5.9.1 Detailed Description

Show a (partial) face expression from a predefined lexicon.

5.9.2 Member Enumeration Documentation

5.9.2.1 Lexeme

```
enum BMLNet.BMLFaceLexeme.Lexeme [strong]
```

5.9.3 Constructor & Destructor Documentation

5.9.3.1 BMLFaceLexeme()

`BMLNet.BMLFaceLexeme.BMLFaceLexeme ()`

constructor

5.9.4 Member Function Documentation

5.9.4.1 Parse()

```
override void BMLNet.BMLFaceLexeme.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: lexeme

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLFace](#).

5.9.5 Member Data Documentation

5.9.5.1 lexeme

[Lexeme](#) `BMLNet.BMLFaceLexeme.lexeme`

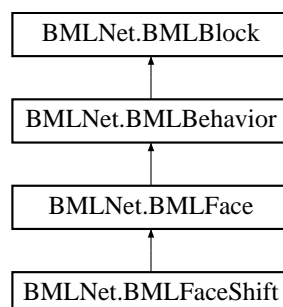
The documentation for this class was generated from the following file:

- `C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLFaceLexeme.cs`

5.10 BMLNet.BMLFaceShift Class Reference

Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit, where the specified compound face expression becomes the new BASE state of the ECAs face.

Inheritance diagram for BMLNet.BMLFaceShift:



Additional Inherited Members

5.10.1 Detailed Description

Compound behavior to specify the timing and alignment of several (partial) face expressions as one unit, where the specified compound face expression becomes the new BASE state of the ECAs face.

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLFaceShift.cs

5.11 BMLNet.BMLFeedback Class Reference

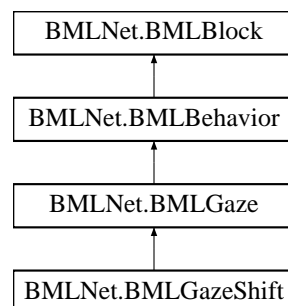
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLFeedback.cs

5.12 BMLNet.BMLGaze Class Reference

Temporarily directs the gaze of the character towards a target. This behavior causes the character to temporarily direct its gaze to the requested target. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.

Inheritance diagram for BMLNet.BMLGaze:



Public Types

- enum **Influence** {
NONE, EYES, HEAD, SHOULDER,
WAIST, WHOLE }
Determines what parts of the body to move to effect the gaze direction.
- enum **Direction** {
RIGHT, LEFT, UP, DOWN,
UPRIGHT, UPLEFT, DOWNLEFT, DOWNRIGHT }
Direction of the offsetDirection angle.

Public Member Functions

- [BMLGaze](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: target, influence, offsetAngle, offsetDirection sync point: start, ready, relax, end

Public Attributes

- string [target](#)
A reference towards a target instance that represents the target direction of the gaze.
- [Influence](#) [influence](#)
Determines what parts of the body to move to effect the gaze direction.
- float [offsetAngle](#)
Adds an angle degrees offset to gaze direction relative to the target in the direction specified in the offsetDirection
- [Direction](#) [offsetDirection](#)
Direction of the offsetDirection angle

Additional Inherited Members

5.12.1 Detailed Description

Temporarily directs the gaze of the character towards a target. This behavior causes the character to temporarily direct its gaze to the requested target. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.

5.12.2 Member Enumeration Documentation

5.12.2.1 Direction

```
enum BMLNet.BMLGaze.Direction [strong]
```

Direction of the offsetDirection angle.

5.12.2.2 Influence

```
enum BMLNet.BMLGaze.Influence [strong]
```

Determines what parts of the body to move to effect the gaze direction.

5.12.3 Constructor & Destructor Documentation

5.12.3.1 BMLGaze()

```
BMLNet.BMLGaze.BMLGaze ( )
```

constructor

5.12.4 Member Function Documentation

5.12.4.1 Parse()

```
override void BMLNet.BMLGaze.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: target, influence, offsetAngle, offsetDirection sync point: start, ready, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

Reimplemented in [BMLNet.BMLGazeShift](#).

5.12.5 Member Data Documentation

5.12.5.1 influence

[Influence](#) `BMLNet.BMLGaze.influence`

Determines what parts of the body to move to effect the gaze direction.

5.12.5.2 offsetAngle

`float BMLNet.BMLGaze.offsetAngle`

Adds an angle degrees offset to gaze direction relative to the target in the direction specified in the offsetDirection

5.12.5.3 offsetDirection

[Direction](#) `BMLNet.BMLGaze.offsetDirection`

Direction of the offsetDirection angle

5.12.5.4 target

`string BMLNet.BMLGaze.target`

A reference towards a target instance that represents the target direction of the gaze.

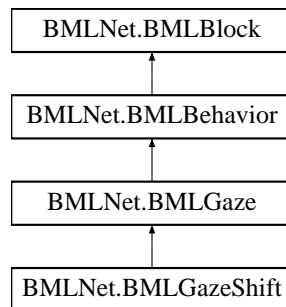
The documentation for this class was generated from the following file:

- `C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLGaze.cs`

5.13 BMLNet.BMLGazeShift Class Reference

Permanently change the gaze direction of the character towards a certain target. This behavior causes the character to direct its gaze to the requested target. This changes the default state of the ECA: after completing this behavior, the new target is the default gaze direction of the ECA. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.

Inheritance diagram for BMLNet.BMLGazeShift:



Public Member Functions

- [BMLGazeShift](#) ()
constructor sync attribute: start, end
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: sync point: start, end

Additional Inherited Members

5.13.1 Detailed Description

Permanently change the gaze direction of the character towards a certain target. This behavior causes the character to direct its gaze to the requested target. This changes the default state of the ECA: after completing this behavior, the new target is the default gaze direction of the ECA. The influence parameter is read as follows: EYE means 'use only the eyes'; HEAD means 'use only head and eyes to change the gaze direction', etcetera.

5.13.2 Constructor & Destructor Documentation

5.13.2.1 BMLGazeShift()

```
BMLNet.BMLGazeShift.BMLGazeShift ( )
```

constructor sync attribute: start, end

5.13.3 Member Function Documentation

5.13.3.1 Parse()

```
override void BMLNet.BMLGazeShift.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: sync point: start, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Reimplemented from [BMLNet.BMLGaze](#).

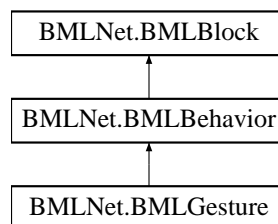
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLGazeShift.cs

5.14 BMLNet.BMLGesture Class Reference

Currently, BML offers two types of gesture behaviors. The first provides a set of gestures recalled by name from a gesticon; the second provides simple pointing gestures. Coordinated movement with arms and hands, recalled from a gesticon by requesting the corresponding lexeme

Inheritance diagram for BMLNet.BMLGesture:



Public Types

- enum [Mode](#) { **NONE**, **LEFT_HAND**, **RIGHT_HAND**, **BOTH_HANDS** }
What hand/arm is being used
- enum [Lexeme](#) { **BEAT** }
Refers to an animation or a controller to realize this particular gesture.

Public Member Functions

- [BMLGesture](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: mode, lexeme sync point: start, ready, strokeStart, stroke, strokeEnd, relax, end

Public Attributes

- [Mode](#) mode
What hand/arm is being used
- [Lexeme](#) lexeme
Refers to an animation or a controller to realize this particular gesture.

Additional Inherited Members

5.14.1 Detailed Description

Currently, BML offers two types of gesture behaviors. The first provides a set of gestures recalled by name from a gesticon; the second provides simple pointing gestures. Coordinated movement with arms and hands, recalled from a gesticon by requesting the corresponding lexeme

5.14.2 Member Enumeration Documentation

5.14.2.1 Lexeme

```
enum BMLNet.BMLGesture.Lexeme [strong]
```

Refers to an animation or a controller to realize this particular gesture.

5.14.2.2 Mode

```
enum BMLNet.BMLGesture.Mode [strong]
```

What hand/arm is being used

5.14.3 Constructor & Destructor Documentation

5.14.3.1 BMLGesture()

```
BMLNet.BMLGesture.BMLGesture ( )
```

constructor

5.14.4 Member Function Documentation

5.14.4.1 Parse()

```
override void BMLNet.BMLGesture.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: mode, lexeme sync point: start, ready, strokeStart, stroke, strokeEnd, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.14.5 Member Data Documentation

5.14.5.1 lexeme

[Lexeme](#) `BMLNet.BMLGesture.lexeme`

Refers to an animation or a controller to realize this particular gesture.

5.14.5.2 mode

[Mode](#) `BMLNet.BMLGesture.mode`

What hand/arm is being used

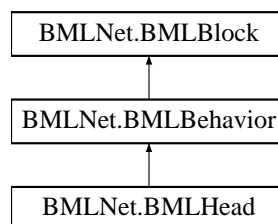
The documentation for this class was generated from the following file:

- `C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLGesture.cs`

5.15 BMLNet.BMLHead Class Reference

Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. The stroke phase of the head motion (from strokeStart till strokeEnd is the "meaningful" part of the head motion. The stroke sync point is the "peak" moment of the motion. If repetition > 1, the meaning of the stroke sync point becomes undefined

Inheritance diagram for BMLNet.BMLHead:



Public Types

- enum [Lexeme](#) { **NONE**, **NOD**, **SHAKE** }
Refers to an animation or a controller to realize this particular head behavior. Minimum set offered by all realizers: [NOD, SHAKE]

Public Member Functions

- [BMLHead](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id, amount, overshoot sync attribute: start, ready, strokeStart, stroke, strokeEnd, relax, end

Public Attributes

- [Lexeme](#) `lexeme`

Refers to an animation or a controller to realize this particular head behavior. Minimum set offered by all realizers: [NOD, SHAKE]

- `int` [repetition](#)

Number of times the basic head motion is repeated.

- `float` [amount](#)

How intense is the head nod? 0 means immeasurable small; 0.5 means "moderate"; 1 means maximally large

Additional Inherited Members

5.15.1 Detailed Description

Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. Movement of the head, recalled from a gesticon by requesting the corresponding lexeme. The stroke phase of the head motion (from `strokeStart` till `strokeEnd` is the "meaningful" part of the head motion. The stroke sync point is the "peak" moment of the motion. If `repetition > 1`, the meaning of the stroke sync point becomes undefined

5.15.2 Member Enumeration Documentation

5.15.2.1 Lexeme

```
enum BMLNet.BMLHead.Lexeme [strong]
```

Refers to an animation or a controller to realize this particular head behavior. Minimum set offered by all realizers: [NOD, SHAKE]

5.15.3 Constructor & Destructor Documentation

5.15.3.1 BMLHead()

```
BMLNet.BMLHead.BMLHead ( )
```

constructor

5.15.4 Member Function Documentation

5.15.4.1 Parse()

```
override void BMLNet.BMLHead.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id, amount, overshoot sync attribute: start, ready, strokeStart, stroke, strokeEnd, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Reimplemented from [BMLNet.BMLBehavior](#).

5.15.5 Member Data Documentation

5.15.5.1 amount

```
float BMLNet.BMLHead.amount
```

How intense is the head nod? 0 means immeasurable small; 0.5 means "moderate"; 1 means maximally large

5.15.5.2 lexeme

```
Lexeme BMLNet.BMLHead.lexeme
```

Refers to an animation or a controller to realize this particular head behavior. Minimum set offered by all realizers: [NOD, SHAKE]

5.15.5.3 repetition

```
int BMLNet.BMLHead.repetition
```

Number of times the basic head motion is repeated.

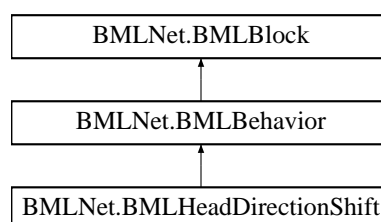
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLHead.cs

5.16 BMLNet.BMLHeadDirectionShift Class Reference

Orient the head towards a target referenced by the target attribute. Permanently orient the head in a certain direction.

Inheritance diagram for BMLNet.BMLHeadDirectionShift:



Public Member Functions

- [BMLHeadDirectionShift](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id, amount, overshoot sync attribute: start, end

Public Attributes

- string [target](#)
target towards which the head is oriented

Additional Inherited Members

5.16.1 Detailed Description

Orient the head towards a target referenced by the target attribute. Permanently orient the head in a certain direction.

5.16.2 Constructor & Destructor Documentation

5.16.2.1 BMLHeadDirectionShift()

```
BMLNet.BMLHeadDirectionShift.BMLHeadDirectionShift ( )
```

constructor

5.16.3 Member Function Documentation

5.16.3.1 Parse()

```
override void BMLNet.BMLHeadDirectionShift.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id, amount, overshoot sync attribute: start, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XmlReader

Reimplemented from [BMLNet.BMLBehavior](#).

5.16.4 Member Data Documentation

5.16.4.1 target

```
string BMLNet.BMLHeadDirectionShift.target
```

target towards which the head is oriented

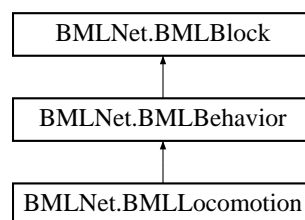
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLHeadDirectionShift.cs

5.17 BMLNet.BMLLocomotion Class Reference

Move the body of the character from one location to another. This behavior causes the character to move to the requested target in the manner described.

Inheritance diagram for BMLNet.BMLLocomotion:



Public Member Functions

- [BMLLocomotion](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing xml attribute: id, target, manner sync point: start, end

Public Attributes

- string [target](#)
A reference towards a target instance that represents the end location of the locomotive behavior.
- string [manner](#)
The general manner of locomotion [WALK, RUN, STRAFE ...] (WALK is the only mandatory element in the set)

Additional Inherited Members

5.17.1 Detailed Description

Move the body of the character from one location to another. This behavior causes the character to move to the requested target in the manner described.

5.17.2 Constructor & Destructor Documentation

5.17.2.1 BMLLocomotion()

```
BMLNet.BMLLocomotion.BMLLocomotion ( )
```

constructor

5.17.3 Member Function Documentation

5.17.3.1 Parse()

```
override void BMLNet.BMLLocomotion.Parse (
    XmlReader reader ) [virtual]
```

parsing xml attribute: id, target, manner sync point: start, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.17.4 Member Data Documentation

5.17.4.1 manner

```
string BMLNet.BMLLocomotion.manner
```

The general manner of locomotion [WALK, RUN, STRAFE ...] (WALK is the only mandatory element in the set)

5.17.4.2 target

```
string BMLNet.BMLLocomotion.target
```

A reference towards a target instance that represents the end location of the locomotive behavior.

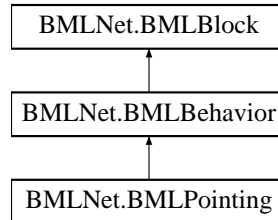
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BML↔Locomotion.cs

5.18 BMLNet.BMLPointing Class Reference

Deictic gesture towards the target specified by the target attribute

Inheritance diagram for BMLNet.BMLPointing:



Public Types

- enum [Mode](#) { **NONE**, **LEFT_HAND**, **RIGHT_HAND**, **BOTH_HANDS** }
What hand/arm is being used

Public Member Functions

- [BMLPointing](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id, target, mode sync point: start, ready, strokeStart, stroke, strokeEnd, relax, end

Public Attributes

- [Mode](#) [mode](#)
What hand/arm is being used
- string [target](#)
The gesture is directed towards this target

Additional Inherited Members

5.18.1 Detailed Description

Deictic gesture towards the target specified by the target attribute

5.18.2 Member Enumeration Documentation

5.18.2.1 Mode

```
enum BMLNet.BMLPointing.Mode [strong]
```

What hand/arm is being used

5.18.3 Constructor & Destructor Documentation

5.18.3.1 BMLPointing()

```
BMLNet.BMLPointing.BMLPointing ( )
```

constructor

5.18.4 Member Function Documentation

5.18.4.1 Parse()

```
override void BMLNet.BMLPointing.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id, target, mode sync point: start, ready, strokeStart, stroke, strokeEnd, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XMLReader

Reimplemented from [BMLNet.BMLBehavior](#).

5.18.5 Member Data Documentation

5.18.5.1 mode

[Mode](#) BMLNet.BMLPointing.mode

What hand/arm is being used

5.18.5.2 target

```
string BMLNet.BMLPointing.target
```

The gesture is directed towards this target

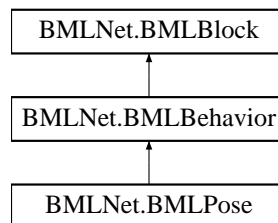
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLPointing.cs

5.19 BMLNet.BMLPose Class Reference

Child element of `<posture>` and `<postureShift>` behaviors, defines additions to the global body posture of the ECA. Child element of `<posture>` and `<postureShift>` behaviors, defines additions that modify the global body posture of the ECA. For each value of the part attribute, only one `<pose>` child is expected to be present. A BML Realizer may define any number of lexemes beyond the ones specified above.

Inheritance diagram for BMLNet.BMLPose:



Public Types

- enum [Part](#) {
NONE, ARMS, LEFT_ARM, RIGHT_ARM, LEGS, LEFT_LEG, RIGHT_LEG, HEAD, WHOLEBODY }
What part of the body is affected? Possible values are [ARMS, LEFT_ARM, RIGHT_ARM, LEGS, LEFT_LEG, RIGHT_LEG, HEAD, WHOLEBODY]
- enum [Lexeme](#) {
NONE, ARMS_AKIMBO, ARMS_CROSSED, ARMS_NEUTRAL, ARMS_OPEN, LEGS_CROSSED, LEGS_NEUTRAL, LEGS_OPEN, LEANING_FORWARD, LEANING_BACKWARD }
What configuration is set to the given part? Some possible values are [ARMS_AKIMBO, ARMS_CROSSED, ARMS_NEUTRAL, ARMS_OPEN, LEGS_CROSSED, LEGS_NEUTRAL, LEGS_OPEN, LEANING_FORWARD, LEANING_BACKWARD, ...]

Public Member Functions

- [BMLPose](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing xml attribute: part, lexeme

Public Attributes

- [Part](#) part
What part of the body is affected? Possible values are [ARMS, LEFT_ARM, RIGHT_ARM, LEGS, LEFT_LEG, RIGHT_LEG, HEAD, WHOLEBODY]
- [Lexeme](#) lexeme
What configuration is set to the given part? Some possible values are [ARMS_AKIMBO, ARMS_CROSSED, ARMS_NEUTRAL, ARMS_OPEN, LEGS_CROSSED, LEGS_NEUTRAL, LEGS_OPEN, LEANING_FORWARD, LEANING_BACKWARD, ...]

Additional Inherited Members

5.19.1 Detailed Description

Child element of <posture> and <postureShift> behaviors, defines additions to the global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines additions that modify the global body posture of the ECA. For each value of the part attribute, only one <pose> child is expected to be present. A BML Realizer may define any number of lexemes beyond the ones specified above.

5.19.2 Member Enumeration Documentation

5.19.2.1 Lexeme

```
enum BMLNet.BMLPose.Lexeme [strong]
```

What configuration is set to the given part? Some possible values are [ARMS_AKIMBO, ARMS_CROSSED, ARMS_NEUTRAL, ARMS_OPEN, LEGS_CROSSED, LEGS_NEUTRAL, LEGS_OPEN, LEANING_FORWARD, LEANING_BACKWARD, ...]

5.19.2.2 Part

```
enum BMLNet.BMLPose.Part [strong]
```

What part of the body is affected? Possible values are [ARMS, LEFT_ARM, RIGHT_ARM, LEGS, LEFT_LEG, RIGHT_LEG, HEAD, WHOLEBODY]

5.19.3 Constructor & Destructor Documentation

5.19.3.1 BMLPose()

```
BMLNet.BMLPose.BMLPose ( )
```

constructor

5.19.4 Member Function Documentation

5.19.4.1 Parse()

```
override void BMLNet.BMLPose.Parse (
    XmlReader reader ) [virtual]
```

parsing xml attribute: part, lexeme

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.19.5 Member Data Documentation

5.19.5.1 lexeme

[Lexeme](#) `BMLNet.BMLPose.lexeme`

What configuration is set to the given part? Some possible values are [ARMS_AKIMBO, ARMS_CROSSED, ARMS_NEUTRAL, ARMS_OPEN, LEGS_CROSSED, LEGS_NEUTRAL, LEGS_OPEN, LEANING_FORWARD, LEANING_BACKWARD, ...]

5.19.5.2 part

[Part](#) `BMLNet.BMLPose.part`

What part of the body is affected? Possible values are [ARMS, LEFT_ARM, RIGHT_ARM, LEGS, LEFT_LEG, RIGHT_LEG, HEAD, WHOLEBODY]

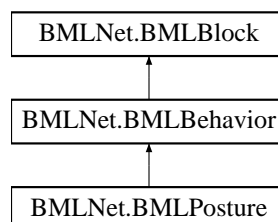
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLPose.cs

5.20 BMLNet.BMLPosture Class Reference

Temporarily change the posture of the ECA. Temporarily change the posture of the ECA. After the <posture> behavior has ended, return to the BASE posture.

Inheritance diagram for BMLNet.BMLPosture:



Public Member Functions

- [BMLPosture](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing xml attribute: id sync point: start, ready, relax, end

Additional Inherited Members

5.20.1 Detailed Description

Temporarily change the posture of the ECA. Temporarily change the posture of the ECA. After the <posture> behavior has ended, return to the BASE posture.

5.20.2 Constructor & Destructor Documentation

5.20.2.1 BMLPosture()

```
BMLNet.BMLPosture.BMLPosture ( )
```

constructor

5.20.3 Member Function Documentation

5.20.3.1 Parse()

```
override void BMLNet.BMLPosture.Parse (
    XmlReader reader ) [virtual]
```

parsing xml attribute: id sync point: start, ready, relax, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XmlReader

Reimplemented from [BMLNet.BMLBehavior](#).

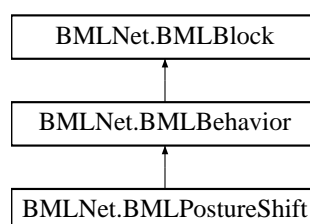
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLPosture.cs

5.21 BMLNet.BMLPostureShift Class Reference

Permanently change the BASE posture of the ECA.

Inheritance diagram for BMLNet.BMLPostureShift:



Public Member Functions

- [BMLPostureShift](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing xml attribute: id sync point: start, end

Additional Inherited Members

5.21.1 Detailed Description

Permanently change the BASE posture of the ECA.

5.21.2 Constructor & Destructor Documentation

5.21.2.1 BMLPostureShift()

```
BMLNet.BMLPostureShift.BMLPostureShift ( )
```

constructor

5.21.3 Member Function Documentation

5.21.3.1 Parse()

```
override void BMLNet.BMLPostureShift.Parse (
    XmlReader reader ) [virtual]
```

parsing xml attribute: id sync point: start, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

XmlReader

Reimplemented from [BMLNet.BMLBehavior](#).

The documentation for this class was generated from the following file:

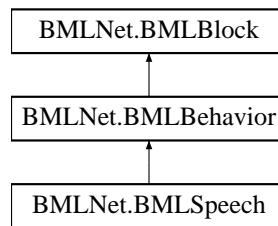
- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLPostureShift.cs

5.22 BMLNet.BMLSpeech Class Reference

Utterance to be spoken by a character. Realization of the <speech> element generates both speech audio (or text) and speech movement, for example using a speech synthesizer and viseme morphing. The<speech> element

requires a sub-element. This sub-element is a `<text>` element that contains the text to be spoken, with optionally embedded `<sync>` elements for alignment with other behaviors.

Inheritance diagram for BMLNet.BMLSpeech:



Public Member Functions

- [BMLSpeech](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml child node: text sync point: start, end

Public Attributes

- string [text](#)
the text that need to be spoken

Additional Inherited Members

5.22.1 Detailed Description

Utterance to be spoken by a character. Realization of the `<speech>` element generates both speech audio (or text) and speech movement, for example using a speech synthesizer and viseme morphing. The `<speech>` element requires a sub-element. This sub-element is a `<text>` element that contains the text to be spoken, with optionally embedded `<sync>` elements for alignment with other behaviors.

5.22.2 Constructor & Destructor Documentation

5.22.2.1 BMLSpeech()

```
BMLNet.BMLSpeech.BMLSpeech ( )
```

constructor

5.22.3 Member Function Documentation

5.22.3.1 Parse()

```
override void BMLNet.BMLSpeech.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml child node: text sync point: start, end

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.22.4 Member Data Documentation

5.22.4.1 text

```
string BMLNet.BMLSpeech.text
```

the text that need to be spoken

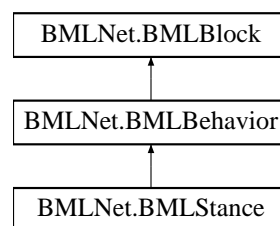
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLSpeech.cs

5.23 BMLNet.BMLStance Class Reference

Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. This global posture may then be modified by one or more <pose> siblings.

Inheritance diagram for BMLNet.BMLStance:



Public Types

- enum [Type](#) { **SITTING**, **CROUCHING**, **STANDING**, **LYING** }
Global body posture. Possible values are [SITTING, CROUCHING, STANDING, LYING]

Public Member Functions

- [BMLStance](#) ()
constructor
- override void [Parse](#) (XmlReader reader)
parsing the xml attribute: id

Public Attributes

- [Type type](#)

Global body posture. Possible values are [SITTING, CROUCHING, STANDING, LYING]

Additional Inherited Members

5.23.1 Detailed Description

Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. Child element of <posture> and <postureShift> behaviors, defines global body posture of the ECA. This global posture may then be modified by one or more <pose> siblings.

5.23.2 Member Enumeration Documentation

5.23.2.1 Type

```
enum BMLNet.BMLStance.Type [strong]
```

Global body posture. Possible values are [SITTING, CROUCHING, STANDING, LYING]

5.23.3 Constructor & Destructor Documentation

5.23.3.1 BMLStance()

```
BMLNet.BMLStance.BMLStance ( )
```

constructor

5.23.4 Member Function Documentation

5.23.4.1 Parse()

```
override void BMLNet.BMLStance.Parse (
    XmlReader reader ) [virtual]
```

parsing the xml attribute: id

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.23.5 Member Data Documentation

5.23.5.1 type

Type `BMLNet.BMLStance.type`

Global body posture. Possible values are [SITTING, CROUCHING, STANDING, LYING]

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLStance.cs

5.24 BMLNet.BMLSyncPoint Class Reference

BML Sync Point class possible format: `behavior_id:sync_id [+/- offset]` A reference to a sync point of another behavior, optionally with a float offset in seconds. By default, this is a behavior in the same `<bml>` block that the `syncref` is contained in; if optional prefix `block_id:` is present, the `syncref` specifies a sync point of a behavior in the `<bml>` block with that ID.) `offset`: A positive float offset in seconds relative to the start time of the surrounding `<bml>` block.

Public Member Functions

- [BMLSyncPoint](#) ([BMLBlock](#) parent, string eventName, string value)
the constructor of BML Sync Point.
- void [Update](#) ([RageBMLNet](#) bmlNet)
function that need to be called everytime the realizer update is called
- bool [TriggerSyncPoint](#) ()
trigger this syncpoint to complete
- bool [IsCompleted](#) ()
is this syncpoint already completed ?
- bool [IsTimerSafe](#) (Dictionary< string, [BMLBlock](#) > blocks, float globalTimer)
function to check whether the timer variable is safe to used or not

5.24.1 Detailed Description

BML Sync Point class possible format: `behavior_id:sync_id [+/- offset]` A reference to a sync point of another behavior, optionally with a float offset in seconds. By default, this is a behavior in the same `<bml>` block that the `syncref` is contained in; if optional prefix `block_id:` is present, the `syncref` specifies a sync point of a behavior in the `<bml>` block with that ID.) `offset`: A positive float offset in seconds relative to the start time of the surrounding `<bml>` block.

5.24.2 Constructor & Destructor Documentation

5.24.2.1 BMLSyncPoint()

```
BMLNet.BMLSyncPoint.BMLSyncPoint (
    BMLBlock parent,
    string eventName,
    string value )
```

the constructor of BML Sync Point.

Parameters

| | |
|------------------|--|
| <i>eventName</i> | |
|------------------|--|

the name of sync point event (start, ready, strokeStart, attackPeak, stroke, strokeEnd, relax, end)

Parameters

| | |
|--------------|--|
| <i>value</i> | |
|--------------|--|

the attribute value that we need to parse.

5.24.3 Member Function Documentation

5.24.3.1 IsCompleted()

```
bool BMLNet.BMLSyncPoint.IsCompleted ( )
```

is this syncpoint already completed ?

Returns

5.24.3.2 IsTimerSafe()

```
bool BMLNet.BMLSyncPoint.IsTimerSafe (
    Dictionary< string, BMLBlock > blocks,
    float globalTimer )
```

function to check whether the timer variable is safe to used or not

Parameters

| | |
|-----------------|--|
| <i>realizer</i> | |
|-----------------|--|

Returns

5.24.3.3 TriggerSyncPoint()

```
bool BMLNet.BMLSyncPoint.TriggerSyncPoint ( )
```

trigger this syncpoint to complete

Returns

5.24.3.4 Update()

```
void BMLNet.BMLSyncPoint.Update (
    RageBMLNet bmlNet )
```

function that need to be called everytime the realizer update is called

Parameters

| | |
|-----------------|--|
| <i>realizer</i> | |
|-----------------|--|

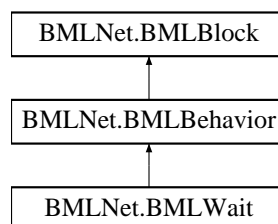
The realizer

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLSyncPoint.cs

5.25 BMLNet.BMLWait Class Reference

Inheritance diagram for BMLNet.BMLWait:



Public Member Functions

- [BMLWait \(\)](#)
constructor
- override void [Parse](#) (XmlReader reader)
parsing the XML attribute: duration

Public Attributes

- float [duration](#)
the duration of the wait in seconds

Additional Inherited Members

5.25.1 Constructor & Destructor Documentation

5.25.1.1 BMLWait()

```
BMLNet.BMLWait.BMLWait ( )
```

constructor

5.25.2 Member Function Documentation

5.25.2.1 Parse()

```
override void BMLNet.BMLWait.Parse (
    XmlReader reader ) [virtual]
```

parsing the XML attribute: duration

Parameters

| | |
|---------------|--|
| <i>reader</i> | |
|---------------|--|

Reimplemented from [BMLNet.BMLBehavior](#).

5.25.3 Member Data Documentation

5.25.3.1 duration

```
float BMLNet.BMLWait.duration
```

the duration of the wait in seconds

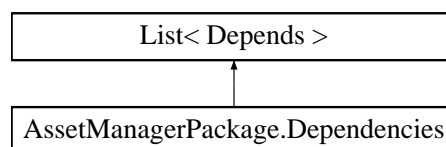
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/BMLNet/BMLWait.cs

5.26 AssetManagerPackage.Dependencies Class Reference

A dependencies.

Inheritance diagram for AssetManagerPackage.Dependencies:



Public Member Functions

- [Dependencies](#) ()

Initializes a new instance of the AssetManagerPackage.dependencies class.

5.26.1 Detailed Description

A dependencies.

5.26.2 Constructor & Destructor Documentation

5.26.2.1 Dependencies()

```
AssetManagerPackage.Dependencies.Dependencies ( )
```

Initializes a new instance of the AssetManagerPackage.dependencies class.

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/Rage↔
VersionInfo.cs

5.27 AssetManagerPackage.Depends Class Reference

A dependency.

Public Member Functions

- [Depends](#) ()
Initializes a new instance of the AssetManagerPackage.Dependency class.

Properties

- String [minVersion](#) [get, set]
Gets or sets the minimum version.
- String [maxVersion](#) [get, set]
Gets or sets the maximum version.
- String [name](#) [get, set]
Gets or sets the name.

5.27.1 Detailed Description

A dependency.

5.27.2 Constructor & Destructor Documentation

5.27.2.1 Depends()

```
AssetManagerPackage.Depends.Depends ( )
```

Initializes a new instance of the AssetManagerPackage.Dependency class.

5.27.3 Property Documentation

5.27.3.1 maxVersion

```
String AssetManagerPackage.Depends.maxVersion [get], [set]
```

Gets or sets the maximum version.

The maximum version.

5.27.3.2 minVersion

```
String AssetManagerPackage.Depends.minVersion [get], [set]
```

Gets or sets the minimum version.

The minimum version.

5.27.3.3 name

```
String AssetManagerPackage.Depends.name [get], [set]
```

Gets or sets the name.

The name.

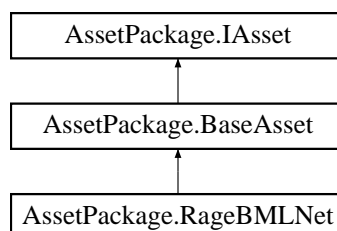
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/Rage↔
VersionInfo.cs

5.28 AssetPackage.IAsset Interface Reference

Interface for asset.

Inheritance diagram for AssetPackage.IAsset:



Properties

- String [Class](#) [get]
Gets the class.
- Dictionary< String, String > [Dependencies](#) [get]
Gets the dependencies.
- String [Id](#) [get]
Gets the identifier.
- String [Maturity](#) [get]
Gets the maturity.
- [ISettings Settings](#) [get, set]
Gets or sets options for controlling the operation.
- String [Version](#) [get]
Gets the version.
- [IBridge Bridge](#) [get, set]
Gets or sets the bridge.

5.28.1 Detailed Description

Interface for asset.

5.28.2 Property Documentation

5.28.2.1 Bridge

[IBridge](#) `AssetPackage.IAsset.Bridge` [get], [set]

Gets or sets the bridge.

The bridge.

5.28.2.2 Class

`String AssetPackage.IAsset.Class` [get]

Gets the class.

The class.

5.28.2.3 Dependencies

`Dictionary<String, String> AssetPackage.IAsset.Dependencies` [get]

Gets the dependencies.

The dependencies (A Dictionary of class=version pairs).

5.28.2.4 Id

```
String AssetPackage.IAsset.Id [get]
```

Gets the identifier.

The identifier.

5.28.2.5 Maturity

```
String AssetPackage.IAsset.Maturity [get]
```

Gets the maturity.

The maturity.

5.28.2.6 Settings

```
ISettings AssetPackage.IAsset.Settings [get], [set]
```

Gets or sets options for controlling the operation.

The settings.

5.28.2.7 Version

```
String AssetPackage.IAsset.Version [get]
```

Gets the version.

The version.

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IAsset.cs

5.29 AssetPackage.IBridge Interface Reference

Interface for bridge.

5.29.1 Detailed Description

Interface for bridge.

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/I↔ Bridge.cs

5.30 AssetPackage.IDataStorage Interface Reference

Interface for data storage.

Public Member Functions

- Boolean [Delete](#) (String fileId)
Deletes the given fileId.
- Boolean [Exists](#) (String fileId)
Check if exists the file with the given identifier.
- String [] [Files](#) ()
Gets the files.
- String [Load](#) (String fileId)
Loads the given file.
- void [Save](#) (String fileId, String fileData)
Saves the given file.

5.30.1 Detailed Description

Interface for data storage.

5.30.2 Member Function Documentation

5.30.2.1 Delete()

```
Boolean AssetPackage.IDataStorage.Delete (
    String fileId )
```

Deletes the given fileId.

Parameters

| | |
|---------------|--------------------------------|
| <i>fileId</i> | The file identifier to delete. |
|---------------|--------------------------------|

Returns

true if it succeeds, false if it fails.

5.30.2.2 Exists()

```
Boolean AssetPackage.IDataStorage.Exists (
    String fileId )
```

Check if exists the file with the given identifier.

Parameters

| | |
|---------------------|--------------------------------|
| <i>file↔ Id</i> | The file identifier to delete. |
|---------------------|--------------------------------|

Returns

true if it succeeds, false if it fails.

5.30.2.3 Files()

```
String [] AssetPackage.IDataStorage.Files ( )
```

Gets the files.

A List<String> gave problems when compiled as PCL and added to a Xamarin Forms project containing iOS, Android and WinPhone subprojects.

Returns

An array of filenames.

5.30.2.4 Load()

```
String AssetPackage.IDataStorage.Load (
    String fileId )
```

Loads the given file.

Parameters

| | |
|---------------------|------------------------------|
| <i>file↔ Id</i> | The file identifier to load. |
|---------------------|------------------------------|

Returns

A String with with the file contents.

5.30.2.5 Save()

```
void AssetPackage.IDataStorage.Save (
    String fileId,
    String fileData )
```

Saves the given file.

Parameters

| | |
|-----------------|----------------------------------|
| <i>fileId</i> | The file identifier to delete. |
| <i>fileData</i> | Information describing the file. |

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IDataStorage.cs

5.31 AssetPackage.IDefaultSettings Interface Reference

Interface for default settings.

This Interface is used to:

- Check if an asset has default (application) settings that override build-in default settings.
- Load these settings from the game environment.
- In certain environments write the actual settings as application defaults. This could for instance be Unity in editor mode.

Public Member Functions

- Boolean [HasDefaultSettings](#) (String Class, String Id)
Query if a 'Class' with Id has default settings.
- String [LoadDefaultSettings](#) (String Class, String Id)
Loads default settings for a 'Class' with Id.
- void [SaveDefaultSettings](#) (String Class, String Id, String fileData)
Saves a default settings for a 'Class' with Id.

5.31.1 Detailed Description

Interface for default settings.

This Interface is used to:

- Check if an asset has default (application) settings that override build-in default settings.
- Load these settings from the game environment.
- In certain environments write the actual settings as application defaults. This could for instance be Unity in editor mode.

Default settings and application default settings are read-only at run- time.

If modification and storage is needed at run-time, the [IDataStorage](#) interface could be used i.c.m. [ISettingsMethods](#).

This interface, if implemented in a bridge, allows to check if an [BaseAsset](#) has some default settings

5.31.2 Member Function Documentation

5.31.2.1 HasDefaultSettings()

```
Boolean AssetPackage.IDefaultSettings.HasDefaultSettings (
    String Class,
    String Id )
```

Query if a 'Class' with Id has default settings.

Parameters

| | |
|--------------|-----------------|
| <i>Class</i> | The class. |
| <i>Id</i> | The identifier. |

Returns

true if default settings, false if not.

5.31.2.2 LoadDefaultSettings()

```
String AssetPackage.IDefaultSettings.LoadDefaultSettings (
    String Class,
    String Id )
```

Loads default settings for a 'Class' with Id.

Note that in Unity the file has to be located in the Resource Directory of the Assets Folder.

Parameters

| | |
|--------------|-----------------|
| <i>Class</i> | The class. |
| <i>Id</i> | The identifier. |

Returns

The default settings.

5.31.2.3 SaveDefaultSettings()

```
void AssetPackage.IDefaultSettings.SaveDefaultSettings (
    String Class,
    String Id,
    String fileData )
```

Saves a default settings for a 'Class' with Id.

This method can only be used during editing the game (so NOT at run-time).

Parameters

| | |
|-----------------|-----------------|
| <i>Class</i> | The class. |
| <i>Id</i> | The identifier. |
| <i>fileData</i> | The File Data. |

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/I↔ DefaultSettings.cs

5.32 AssetPackage.ILog Interface Reference

Interface for logger.

Public Member Functions

- void [Log](#) ([Severity](#) severity, String msg)
Executes the log operation.

5.32.1 Detailed Description

Interface for logger.

5.32.2 Member Function Documentation

5.32.2.1 Log()

```
void AssetPackage.ILog.Log (
    Severity severity,
    String msg )
```

Executes the log operation.

Implement this in Game Engine Code.

Parameters

| | |
|-----------------|---------------|
| <i>severity</i> | The severity. |
| <i>msg</i> | The message. |

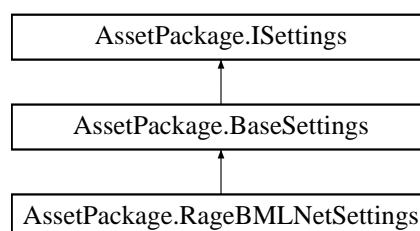
The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/ILog.cs

5.33 AssetPackage.ISettings Interface Reference

Interface for settings.

Inheritance diagram for AssetPackage.ISettings:



5.33.1 Detailed Description

Interface for settings.

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/ISettings.cs

5.34 AssetPackage.IWebServiceRequest Interface Reference

Interface for web service request.

Public Member Functions

- void [WebServiceRequest](#) ([RequestSettings](#) requestSettings, out [RequestResponse](#) requestResponse)
Web service request.

5.34.1 Detailed Description

Interface for web service request.

Implemented on a Bridge.

5.34.2 Member Function Documentation

5.34.2.1 WebServiceRequest()

```
void AssetPackage.IWebServiceRequest.WebServiceRequest (
    RequestSettings requestSettings,
    out RequestResponse requestResponse )
```

Web service request.

Returns

A [RequestResponse](#).

Parameters

| | |
|------------------------|--|
| <i>requestSettings</i> | Options for controlling the operation. |
| <i>requestResponse</i> | The request response. |

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IWebServiceRequest.cs

5.35 AssetPackage.IWebServiceRequestAsync Interface Reference

Interface for web service request.

Public Member Functions

- void [WebServiceRequestAsync](#) (string method, Uri uri, Dictionary< string, string > headers, string body, [IWebServiceResponseAsync](#) response)
Web service request.

5.35.1 Detailed Description

Interface for web service request.

Implemented on a Bridge. Will be replaced by the code from IWebServiceRequest2 once tested.

5.35.2 Member Function Documentation

5.35.2.1 WebServiceRequestAsync()

```
void AssetPackage.IWebServiceRequestAsync.WebServiceRequestAsync (
    string method,
    Uri uri,
    Dictionary< string, string > headers,
    string body,
    IWebServiceResponseAsync response )
```

Web service request.

Parameters

| | |
|-----------------|----------------------|
| <i>method</i> | The method. |
| <i>uri</i> | URI of the document. |
| <i>headers</i> | The headers. |
| <i>body</i> | The body. |
| <i>response</i> | The response. |

The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IWebServiceRequestAsync.cs

5.36 AssetPackage.IWebServiceResponseAsync Interface Reference

Interface for web service response.

Public Member Functions

- void [Error](#) (string url, string msg)
Called when a WebRequest results in an Error.
- void [Success](#) (string url, int code, Dictionary< string, string > headers, string body)
Called after a Successfull WebRequest (no Exceptions).

5.36.1 Detailed Description

Interface for web service response.

Implemented by assets requesting result notification of a [IWebServiceRequest](#).

5.36.2 Member Function Documentation

5.36.2.1 Error()

```
void AssetPackage.IWebServiceResponseAsync.Error (
    string url,
    string msg )
```

Called when a WebRequest results in an Error.

Parameters

| | |
|------------|----------------------|
| <i>url</i> | URL of the document. |
| <i>msg</i> | The error message. |

5.36.2.2 Success()

```
void AssetPackage.IWebServiceResponseAsync.Success (
    string url,
    int code,
    Dictionary< string, string > headers,
    string body )
```

Called after a Successfull WebRequest (no Exceptions).

Parameters

| | |
|----------------|----------------------|
| <i>url</i> | URL of the document. |
| <i>code</i> | The code. |
| <i>headers</i> | The headers. |
| <i>body</i> | The body. |

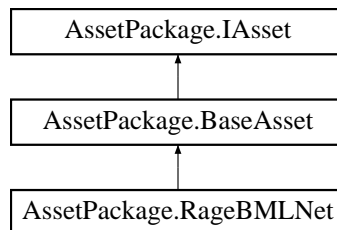
The documentation for this interface was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IWebServiceRequestAsync.cs

5.37 AssetPackage.RageBMLNet Class Reference

An [BMLNet](#) Rage asset

Inheritance diagram for AssetPackage.RageBMLNet:



Public Member Functions

- delegate void [SyncPointCompleted](#) (string id, string eventName)
callback function. it will be called when the specific sync point is completed
- [RageBMLNet](#) ()
Initializes a new instance of the RageBMLNet.Asset class.
- void **ParseFromFile** (string filename)
- void **ParseFromString** (string xml)
- void [Update](#) (float deltaTime)
update function will be called everytime when the program is run. it can be called inside Unity Update function
- void [TriggerSyncPoint](#) (string id, string eventName)
this function can be called from outside library to trigger sync point.
- [BMLBlock GetBehaviorFromId](#) (string id)
function to get behavior from ID

Public Attributes

- [SyncPointCompleted](#) **OnSyncPointCompleted**

Properties

- override [ISettings Settings](#) [get, set]
Gets or sets options for controlling the operation.
- Dictionary< string, [BMLBlock](#) > [ScheduledBlocks](#) [get]
the dictionary that hold the blocks / behavior that need to be run
- float [Timer](#) [get]
global timer

Additional Inherited Members

5.37.1 Detailed Description

An [BMLNet](#) Rage asset

5.37.2 Constructor & Destructor Documentation

5.37.2.1 RageBMLNet()

```
AssetPackage.RageBMLNet.RageBMLNet ( )
```

Initializes a new instance of the RageBMLNet.Asset class.

Create Settings and let it's [BaseSettings](#) class assign Defaultvalues where it can.

5.37.3 Member Function Documentation

5.37.3.1 GetBehaviorFromId()

```
BMLBlock AssetPackage.RageBMLNet.GetBehaviorFromId (
    string id )
```

function to get behavior from ID

Parameters

| | |
|-----------|--|
| <i>id</i> | |
|-----------|--|

Returns

5.37.3.2 SyncPointCompleted()

```
delegate void AssetPackage.RageBMLNet.SyncPointCompleted (
    string id,
    string eventName )
```

callback function. it will be called when the specific sync point is completed

Parameters

| | |
|-----------|--|
| <i>id</i> | |
|-----------|--|

the ID of block

Parameters

| | |
|------------------|--|
| <i>eventName</i> | |
|------------------|--|

the event name of sync point (start, ready, strokeStart, attackPeak, stroke, strokeEnd, relax, end)

5.37.3.3 TriggerSyncPoint()

```
void AssetPackage.RageBMLNet.TriggerSyncPoint (
    string id,
    string eventName )
```

this function can be called from outside library to trigger sync point.

Parameters

| | |
|-----------|--|
| <i>id</i> | |
|-----------|--|

the ID of the block where the sync point is resided

Parameters

| | |
|------------------|--|
| <i>eventName</i> | |
|------------------|--|

the event name of sync point (start, ready, strokeStart, attackPeak, stroke, strokeEnd, relax, end)

5.37.3.4 Update()

```
void AssetPackage.RageBMLNet.Update (
    float deltaTime )
```

update function will be called everytime when the program is run. it can be called inside Unity Update function

Parameters

| | |
|------------------|--|
| <i>deltaTime</i> | |
|------------------|--|

the time from last called

5.37.4 Property Documentation

5.37.4.1 ScheduledBlocks

```
Dictionary<string, BMLBlock> AssetPackage.RageBMLNet.ScheduledBlocks [get]
```

the dictionary that hold the blocks / behavior that need to be run

5.37.4.2 Settings

```
override ISettings AssetPackage.RageBMLNet.Settings [get], [set]
```

Gets or sets options for controlling the operation.

Besides the toXml() and fromXml() methods, we never use this property but use it's correctly typed backing field 'settings' instead.

This property should go into each asset having Settings of its own.

The actual class used should be derived from [BaseAsset](#) (and not directly from ISetting).

The settings.

5.37.4.3 Timer

```
float AssetPackage.RageBMLNet.Timer [get]
```

global timer

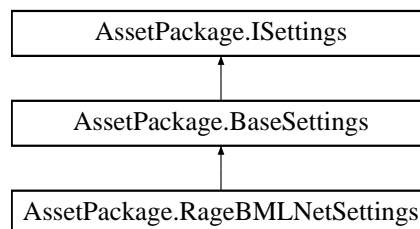
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/RageBMLNet.cs

5.38 AssetPackage.RageBMLNetSettings Class Reference

An asset settings.

Inheritance diagram for AssetPackage.RageBMLNetSettings:



Public Member Functions

- [RageBMLNetSettings](#) ()
Initializes a new instance of the RageBMLNet.AssetSettings class.

Additional Inherited Members

5.38.1 Detailed Description

An asset settings.

[BaseSettings](#) contains the (de-)serialization methods.

5.38.2 Constructor & Destructor Documentation

5.38.2.1 RageBMLNetSettings()

```
AssetPackage.RageBMLNetSettings.RageBMLNetSettings ( )
```

Initializes a new instance of the RageBMLNet.AssetSettings class.

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageBMLNet/RageBMLNetSettings.cs

5.39 AssetManagerPackage.RageVersionInfo Class Reference

Information about the rage version.

Public Member Functions

- [RageVersionInfo](#) ()
Initializes a new instance of the [AssetManagerPackage.RageVersionInfo](#) class.
- String [SaveVersionInfo](#) ()
Saves the version information.

Static Public Member Functions

- static [RageVersionInfo LoadVersionInfo](#) (String xml)
Loads version information.

Properties

- String [Id](#) [get, set]
Gets or sets the identifier.
- Int32 [Major](#) [get, set]
Gets or sets the major.
- Int32 [Minor](#) [get, set]
Gets or sets the minor.
- Int32 [Build](#) [get, set]
Gets or sets the build.
- Int32 [Revision](#) [get, set]
Gets or sets the revision.
- String [Maturity](#) [get, set]
Gets or sets the maturity.
- [Dependencies Dependencies](#) [get, set]
Gets or sets the dependencies.

5.39.1 Detailed Description

Information about the rage version.

5.39.2 Constructor & Destructor Documentation

5.39.2.1 RageVersionInfo()

```
AssetManagerPackage.RageVersionInfo.RageVersionInfo ( )
```

Initializes a new instance of the [AssetManagerPackage.RageVersionInfo](#) class.

5.39.3 Member Function Documentation

5.39.3.1 LoadVersionInfo()

```
static RageVersionInfo AssetManagerPackage.RageVersionInfo.LoadVersionInfo (
    String xml ) [static]
```

Loads version information.

Parameters

| | |
|------------|----------|
| <i>xml</i> | The XML. |
|------------|----------|

Use DataContractSerializer or DataContractJsonSerializer?

5.39.3.2 SaveVersionInfo()

```
String AssetManagerPackage.RageVersionInfo.SaveVersionInfo ( )
```

Saves the version information.

Returns

A String.

Use DataContractSerializer or DataContractJsonSerializer?

5.39.4 Property Documentation

5.39.4.1 Build

```
Int32 AssetManagerPackage.RageVersionInfo.Build [get], [set]
```

Gets or sets the build.

The build.

5.39.4.2 Dependencies

`Dependencies` AssetManagerPackage.RageVersionInfo.Dependencies [get], [set]

Gets or sets the dependencies.

The dependencies.

5.39.4.3 Id

`String` AssetManagerPackage.RageVersionInfo.Id [get], [set]

Gets or sets the identifier.

The identifier.

5.39.4.4 Major

`Int32` AssetManagerPackage.RageVersionInfo.Major [get], [set]

Gets or sets the major.

The major.

5.39.4.5 Maturity

`String` AssetManagerPackage.RageVersionInfo.Maturity [get], [set]

Gets or sets the maturity.

The maturity.

5.39.4.6 Minor

`Int32` AssetManagerPackage.RageVersionInfo.Minor [get], [set]

Gets or sets the minor.

The minor.

5.39.4.7 Revision

`Int32` AssetManagerPackage.RageVersionInfo.Revision [get], [set]

Gets or sets the revision.

The revision.

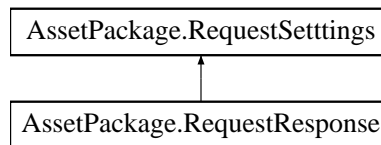
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/RageVersionInfo.cs

5.40 AssetPackage.RequestResponse Class Reference

Response results.

Inheritance diagram for AssetPackage.RequestResponse:



Public Member Functions

- [RequestResponse](#) ()
Initializes a new instance of the [AssetPackage.RequestResponse](#) class.
- [RequestResponse](#) ([RequestSettings](#) settings)
Initializes a new instance of the [AssetPackage.RequestResponse](#) class.

Public Attributes

- int [responseCode](#)
The response code.
- string [responsMessage](#)
Message describing the respons.
- Dictionary< String, String > [responseHeaders](#)
The response headers.

Properties

- bool [ResultAllowed](#) [get]
Gets a value indicating whether result is allowed.

5.40.1 Detailed Description

Response results.

5.40.2 Constructor & Destructor Documentation

5.40.2.1 RequestResponse() [1/2]

```
AssetPackage.RequestResponse.RequestResponse ( )
```

Initializes a new instance of the [AssetPackage.RequestResponse](#) class.

5.40.2.2 RequestResponse() [2/2]

```
AssetPackage.RequestResponse.RequestResponse (
    RequestSettings settings )
```

Initializes a new instance of the [AssetPackage.RequestResponse](#) class.

The body is not copied as it will contain thee response body instead.

Parameters

| | |
|-----------------|--|
| <i>settings</i> | Options for controlling the operation. |
|-----------------|--|

5.40.3 Member Data Documentation

5.40.3.1 responseCode

```
int AssetPackage.RequestResponse.responseCode
```

The response code.

5.40.3.2 responseHeaders

```
Dictionary<String, String> AssetPackage.RequestResponse.responseHeaders
```

The response headers.

5.40.3.3 responsMessage

```
string AssetPackage.RequestResponse.responsMessage
```

Message describing the respons.

5.40.4 Property Documentation

5.40.4.1 ResultAllowed

```
bool AssetPackage.RequestResponse.ResultAllowed [get]
```

Gets a value indicating whether result is allowed.

true if result allowed, false if not.

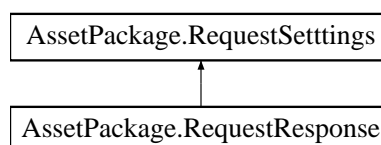
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IWeb↔ServiceRequest.cs

5.41 AssetPackage.RequestSettings Class Reference

Request Settings.

Inheritance diagram for AssetPackage.RequestSettings:



Public Member Functions

- [RequestSettings](#) ()

Initializes a new instance of the `AssetPackage.requestParameters` class.

Public Attributes

- string [method](#)
The method.
- Uri [uri](#)
URI of the document.
- Dictionary< String, String > [requestHeaders](#)
The request headers.
- String [body](#)
The body.
- List< int > [allowedResponsCodes](#)
The allowed responses.

5.41.1 Detailed Description

Request Settings.

5.41.2 Constructor & Destructor Documentation

5.41.2.1 RequestSettings()

```
AssetPackage.RequestSettings.RequestSettings ( )
```

Initializes a new instance of the `AssetPackage.requestParameters` class.

5.41.3 Member Data Documentation

5.41.3.1 allowedResponsCodes

```
List<int> AssetPackage.RequestSettings.allowedResponsCodes
```

The allowed responses.

5.41.3.2 body

```
String AssetPackage.RequestSettings.body
```

The body.

5.41.3.3 method

```
string AssetPackage.RequestSettings.method
```

The method.

5.41.3.4 requestHeaders

```
Dictionary<String, String> AssetPackage.RequestSettings.requestHeaders
```

The request headers.

5.41.3.5 uri

```
Uri AssetPackage.RequestSettings.uri
```

URI of the document.

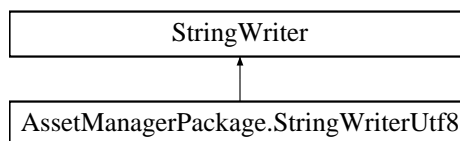
The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/IWeb↔ServiceRequest.cs

5.42 AssetManagerPackage.StringWriterUtf8 Class Reference

A StringWriter UTF8.

Inheritance diagram for AssetManagerPackage.StringWriterUtf8:



Properties

- override Encoding [Encoding](#) [get]
Gets the T:System.Text.Encoding in which the output is written.

5.42.1 Detailed Description

A StringWriter UTF8.

Fix-up for XDocument Serialization defaulting to utf-16.

5.42.2 Property Documentation

5.42.2.1 Encoding

```
override Encoding AssetManagerPackage.StringWriterUtf8.Encoding [get]
```

Gets the T:System.Text.Encoding in which the output is written.

The Encoding in which the output is written.

The documentation for this class was generated from the following file:

- C:/Users/Chris021/Google Drive/phd/projects/Virtual Human/BMLNet.git/trunk/RageAssetManager/Rage↔Extensions.cs

Index

- allowedResponsCodes
 - AssetPackage::RequestSettings, 86
- amount
 - BMLNet::BMLFace, 33
 - BMLNet::BMLHead, 46
- AssetManagerPackage, 9
- AssetManagerPackage.AssetManager, 13
- AssetManagerPackage.Dependencies, 63
- AssetManagerPackage.Depends, 64
- AssetManagerPackage.RageVersionInfo, 81
- AssetManagerPackage.StringWriterUtf8, 87
- AssetManagerPackage::AssetManager
 - Bridge, 16
 - findAssetByClass, 14
 - findAssetById, 14
 - findAssetsByClass, 14
 - getInterface< T >, 15
 - Instance, 16
 - Log, 15
 - registerAssetInstance, 15
 - VersionAndDependenciesReport, 16
- AssetManagerPackage::Dependencies
 - Dependencies, 64
- AssetManagerPackage::Depends
 - Depends, 64
 - maxVersion, 65
 - minVersion, 65
 - name, 65
- AssetManagerPackage::RageVersionInfo
 - Build, 82
 - Dependencies, 82
 - Id, 83
 - LoadVersionInfo, 82
 - Major, 83
 - Maturity, 83
 - Minor, 83
 - RageVersionInfo, 82
 - Revision, 83
 - SaveVersionInfo, 82
- AssetManagerPackage::StringWriterUtf8
 - Encoding, 87
- AssetPackage, 10
 - LogLevel, 11
 - Severity, 11
- AssetPackage.BaseAsset, 17
- AssetPackage.BaseSettings, 23
- AssetPackage.IAsset, 65
- AssetPackage.IBridge, 67
- AssetPackage.IDataStorage, 68
 - AssetPackage.IDefaultSettings, 70
 - AssetPackage.ILog, 73
 - AssetPackage.ISettings, 73
 - AssetPackage.IWebServiceRequest, 74
 - AssetPackage.IWebServiceRequestAsync, 75
 - AssetPackage.IWebServiceResponseAsync, 76
 - AssetPackage.RageBMLNet, 77
 - AssetPackage.RageBMLNetSettings, 80
 - AssetPackage.RequestResponse, 84
 - AssetPackage.RequestSettings, 85
- AssetPackage::BaseAsset
 - BaseAsset, 18
 - Bridge, 22
 - Class, 22
 - Dependencies, 22
 - GetEmbeddedResource, 19
 - getInterface< T >, 19
 - hasSettings, 22
 - Id, 22
 - LoadDefaultSettings, 19
 - LoadSettings, 19
 - Log, 20
 - Maturity, 22
 - SaveDefaultSettings, 20
 - SaveSettings, 21
 - Settings, 23
 - SettingsFromXml, 21
 - SettingsToXml, 21
 - Version, 23
 - VersionInfo, 23
- AssetPackage::BaseSettings
 - BaseSettings, 24
 - UpdateDefaultValues, 24
- AssetPackage::IAsset
 - Bridge, 66
 - Class, 66
 - Dependencies, 66
 - Id, 66
 - Maturity, 67
 - Settings, 67
 - Version, 67
- AssetPackage::IDataStorage
 - Delete, 68
 - Exists, 68
 - Files, 69
 - Load, 69
 - Save, 69
- AssetPackage::IDefaultSettings
 - HasDefaultSettings, 71

- LoadDefaultSettings, [72](#)
- SaveDefaultSettings, [72](#)
- AssetPackage::ILog
 - Log, [73](#)
- AssetPackage::IWebRequest
 - WebRequest, [74](#)
- AssetPackage::IWebRequestAsync
 - WebRequestAsync, [75](#)
- AssetPackage::IWebResponseAsync
 - Error, [76](#)
 - Success, [76](#)
- AssetPackage::RageBMLNet
 - GetBehaviorFromId, [78](#)
 - RageBMLNet, [78](#)
 - ScheduledBlocks, [79](#)
 - Settings, [79](#)
 - SyncPointCompleted, [78](#)
 - Timer, [80](#)
 - TriggerSyncPoint, [79](#)
 - Update, [79](#)
- AssetPackage::RageBMLNetSettings
 - RageBMLNetSettings, [81](#)
- AssetPackage::RequestResponse
 - RequestResponse, [84](#)
 - responsMessage, [85](#)
 - responseCode, [85](#)
 - responseHeaders, [85](#)
 - ResultAllowed, [85](#)
- AssetPackage::RequestSettings
 - allowedResponsCodes, [86](#)
 - body, [86](#)
 - method, [86](#)
 - requestHeaders, [87](#)
 - RequestSettings, [86](#)
 - uri, [87](#)
- au
 - BMLNet::BMLFaceFacs, [35](#)
- BMLBehavior
 - BMLNet::BMLBehavior, [25](#)
- BMLBlock
 - BMLNet::BMLBlock, [28](#)
- BMLBml
 - BMLNet::BMLBml, [31](#)
- BMLFace
 - BMLNet::BMLFace, [33](#)
- BMLFaceFacs
 - BMLNet::BMLFaceFacs, [35](#)
- BMLFaceLexeme
 - BMLNet::BMLFaceLexeme, [37](#)
- BMLGaze
 - BMLNet::BMLGaze, [39](#)
- BMLGazeShift
 - BMLNet::BMLGazeShift, [41](#)
- BMLGesture
 - BMLNet::BMLGesture, [43](#)
- BMLHead
 - BMLNet::BMLHead, [45](#)
- BMLHeadDirectionShift
 - BMLNet::BMLHeadDirectionShift, [47](#)
- BMLLocomotion
 - BMLNet::BMLLocomotion, [49](#)
- BMLNet, [11](#)
- BMLNet.BMLBehavior, [24](#)
- BMLNet.BMLBlock, [26](#)
- BMLNet.BMLBml, [30](#)
- BMLNet.BMLFace, [32](#)
- BMLNet.BMLFaceFacs, [34](#)
- BMLNet.BMLFaceLexeme, [36](#)
- BMLNet.BMLFaceShift, [37](#)
- BMLNet.BMLFeedback, [38](#)
- BMLNet.BMLGaze, [38](#)
- BMLNet.BMLGazeShift, [41](#)
- BMLNet.BMLGesture, [42](#)
- BMLNet.BMLHead, [44](#)
- BMLNet.BMLHeadDirectionShift, [46](#)
- BMLNet.BMLLocomotion, [48](#)
- BMLNet.BMLPointing, [50](#)
- BMLNet.BMLPose, [52](#)
- BMLNet.BMLPosture, [54](#)
- BMLNet.BMLPostureShift, [55](#)
- BMLNet.BMLSpeech, [56](#)
- BMLNet.BMLStance, [58](#)
- BMLNet.BMLSyncPoint, [60](#)
- BMLNet.BMLWait, [62](#)
- BMLNet::BMLBehavior
 - BMLBehavior, [25](#)
 - Parse, [26](#)
- BMLNet::BMLBlock
 - BMLBlock, [28](#)
 - id, [30](#)
 - parentBml, [30](#)
 - Parse, [28](#)
 - syncPoints, [30](#)
 - TryParseAttribute< T >, [28](#)
 - TryParseSyncPoint, [29](#)
- BMLNet::BMLBml
 - BMLBml, [31](#)
 - characterId, [32](#)
 - Composition, [31](#)
 - composition, [32](#)
 - Parse, [31](#)
- BMLNet::BMLFace
 - amount, [33](#)
 - BMLFace, [33](#)
 - overshoot, [33](#)
 - Parse, [33](#)
- BMLNet::BMLFaceFacs
 - au, [35](#)
 - BMLFaceFacs, [35](#)
 - Side, [35](#)
 - side, [35](#)
- BMLNet::BMLFaceLexeme
 - BMLFaceLexeme, [37](#)
 - Lexeme, [36](#)
 - lexeme, [37](#)
 - Parse, [37](#)

- BMLNet::BMLGaze
 - BMLGaze, [39](#)
 - Direction, [39](#)
 - Influence, [39](#)
 - influence, [40](#)
 - offsetAngle, [40](#)
 - offsetDirection, [40](#)
 - Parse, [39](#)
 - target, [40](#)
- BMLNet::BMLGazeShift
 - BMLGazeShift, [41](#)
 - Parse, [41](#)
- BMLNet::BMLGesture
 - BMLGesture, [43](#)
 - Lexeme, [43](#)
 - lexeme, [44](#)
 - Mode, [43](#)
 - mode, [44](#)
 - Parse, [43](#)
- BMLNet::BMLHead
 - amount, [46](#)
 - BMLHead, [45](#)
 - Lexeme, [45](#)
 - lexeme, [46](#)
 - Parse, [45](#)
 - repetition, [46](#)
- BMLNet::BMLHeadDirectionShift
 - BMLHeadDirectionShift, [47](#)
 - Parse, [47](#)
 - target, [47](#)
- BMLNet::BMLLocomotion
 - BMLLocomotion, [49](#)
 - manner, [49](#)
 - Parse, [49](#)
 - target, [49](#)
- BMLNet::BMLPointing
 - BMLPointing, [51](#)
 - Mode, [50](#)
 - mode, [51](#)
 - Parse, [51](#)
 - target, [51](#)
- BMLNet::BMLPose
 - BMLPose, [53](#)
 - Lexeme, [53](#)
 - lexeme, [54](#)
 - Parse, [53](#)
 - Part, [53](#)
 - part, [54](#)
- BMLNet::BMLPosture
 - BMLPosture, [55](#)
 - Parse, [55](#)
- BMLNet::BMLPostureShift
 - BMLPostureShift, [56](#)
 - Parse, [56](#)
- BMLNet::BMLSpeech
 - BMLSpeech, [57](#)
 - Parse, [57](#)
 - text, [58](#)
- BMLNet::BMLStance
 - BMLStance, [59](#)
 - Parse, [59](#)
 - Type, [59](#)
 - type, [60](#)
- BMLNet::BMLSyncPoint
 - BMLSyncPoint, [60](#)
 - IsCompleted, [61](#)
 - IsTimerSafe, [61](#)
 - TriggerSyncPoint, [61](#)
 - Update, [61](#)
- BMLNet::BMLWait
 - BMLWait, [62](#)
 - duration, [63](#)
 - Parse, [63](#)
- BMLPointing
 - BMLNet::BMLPointing, [51](#)
- BMLPose
 - BMLNet::BMLPose, [53](#)
- BMLPosture
 - BMLNet::BMLPosture, [55](#)
- BMLPostureShift
 - BMLNet::BMLPostureShift, [56](#)
- BMLSpeech
 - BMLNet::BMLSpeech, [57](#)
- BMLStance
 - BMLNet::BMLStance, [59](#)
- BMLSyncPoint
 - BMLNet::BMLSyncPoint, [60](#)
- BMLWait
 - BMLNet::BMLWait, [62](#)
- BaseAsset
 - AssetPackage::BaseAsset, [18](#)
- BaseSettings
 - AssetPackage::BaseSettings, [24](#)
- body
 - AssetPackage::RequestSettings, [86](#)
- Bridge
 - AssetManagerPackage::AssetManager, [16](#)
 - AssetPackage::BaseAsset, [22](#)
 - AssetPackage::IAsset, [66](#)
- Build
 - AssetManagerPackage::RageVersionInfo, [82](#)
- characterId
 - BMLNet::BMLBml, [32](#)
- Class
 - AssetPackage::BaseAsset, [22](#)
 - AssetPackage::IAsset, [66](#)
- Composition
 - BMLNet::BMLBml, [31](#)
- composition
 - BMLNet::BMLBml, [32](#)
- Delete
 - AssetPackage::IDataStorage, [68](#)
- Dependencies
 - AssetManagerPackage::Dependencies, [64](#)
 - AssetManagerPackage::RageVersionInfo, [82](#)

- AssetPackage::BaseAsset, 22
- AssetPackage::IAsset, 66
- Depends
 - AssetManagerPackage::Depends, 64
- Direction
 - BMLNet::BMLGaze, 39
- duration
 - BMLNet::BMLWait, 63
- Encoding
 - AssetManagerPackage::StringWriterUtf8, 87
- Error
 - AssetPackage::IWebServiceResponseAsync, 76
- Exists
 - AssetPackage::IDataStorage, 68
- Files
 - AssetPackage::IDataStorage, 69
- findAssetByClass
 - AssetManagerPackage::AssetManager, 14
- findAssetById
 - AssetManagerPackage::AssetManager, 14
- findAssetsByClass
 - AssetManagerPackage::AssetManager, 14
- GetBehaviorFromId
 - AssetPackage::RageBMLNet, 78
- GetEmbeddedResource
 - AssetPackage::BaseAsset, 19
- getInterface< T >
 - AssetManagerPackage::AssetManager, 15
 - AssetPackage::BaseAsset, 19
- HasDefaultSettings
 - AssetPackage::IDefaultSettings, 71
- hasSettings
 - AssetPackage::BaseAsset, 22
- Id
 - AssetManagerPackage::RageVersionInfo, 83
 - AssetPackage::BaseAsset, 22
 - AssetPackage::IAsset, 66
- id
 - BMLNet::BMLBlock, 30
- Influence
 - BMLNet::BMLGaze, 39
- influence
 - BMLNet::BMLGaze, 40
- Instance
 - AssetManagerPackage::AssetManager, 16
- IsCompleted
 - BMLNet::BMLSyncPoint, 61
- IsTimerSafe
 - BMLNet::BMLSyncPoint, 61
- Lexeme
 - BMLNet::BMLFaceLexeme, 36
 - BMLNet::BMLGesture, 43
 - BMLNet::BMLHead, 45
 - BMLNet::BMLPose, 53
- lexeme
 - BMLNet::BMLFaceLexeme, 37
 - BMLNet::BMLGesture, 44
 - BMLNet::BMLHead, 46
 - BMLNet::BMLPose, 54
- Load
 - AssetPackage::IDataStorage, 69
- LoadDefaultSettings
 - AssetPackage::BaseAsset, 19
 - AssetPackage::IDefaultSettings, 72
- LoadSettings
 - AssetPackage::BaseAsset, 19
- LoadVersionInfo
 - AssetManagerPackage::RageVersionInfo, 82
- Log
 - AssetManagerPackage::AssetManager, 15
 - AssetPackage::BaseAsset, 20
 - AssetPackage::ILog, 73
- LogLevel
 - AssetPackage, 11
- Major
 - AssetManagerPackage::RageVersionInfo, 83
- manner
 - BMLNet::BMLLocomotion, 49
- Maturity
 - AssetManagerPackage::RageVersionInfo, 83
 - AssetPackage::BaseAsset, 22
 - AssetPackage::IAsset, 67
- maxVersion
 - AssetManagerPackage::Depends, 65
- method
 - AssetPackage::RequestSettings, 86
- minVersion
 - AssetManagerPackage::Depends, 65
- Minor
 - AssetManagerPackage::RageVersionInfo, 83
- Mode
 - BMLNet::BMLGesture, 43
 - BMLNet::BMLPointing, 50
- mode
 - BMLNet::BMLGesture, 44
 - BMLNet::BMLPointing, 51
- name
 - AssetManagerPackage::Depends, 65
- offsetAngle
 - BMLNet::BMLGaze, 40
- offsetDirection
 - BMLNet::BMLGaze, 40
- overshoot
 - BMLNet::BMLFace, 33
- parentBml
 - BMLNet::BMLBlock, 30
- Parse
 - BMLNet::BMLBehavior, 26
 - BMLNet::BMLBlock, 28

- BMLNet::BMLBml, 31
- BMLNet::BMLFace, 33
- BMLNet::BMLFaceLexeme, 37
- BMLNet::BMLGaze, 39
- BMLNet::BMLGazeShift, 41
- BMLNet::BMLGesture, 43
- BMLNet::BMLHead, 45
- BMLNet::BMLHeadDirectionShift, 47
- BMLNet::BMLLocomotion, 49
- BMLNet::BMLPointing, 51
- BMLNet::BMLPose, 53
- BMLNet::BMLPosture, 55
- BMLNet::BMLPostureShift, 56
- BMLNet::BMLSpeech, 57
- BMLNet::BMLStance, 59
- BMLNet::BMLWait, 63
- Part
 - BMLNet::BMLPose, 53
- part
 - BMLNet::BMLPose, 54
- RageBMLNet
 - AssetPackage::RageBMLNet, 78
- RageBMLNetSettings
 - AssetPackage::RageBMLNetSettings, 81
- RageVersionInfo
 - AssetManagerPackage::RageVersionInfo, 82
- registerAssetInstance
 - AssetManagerPackage::AssetManager, 15
- repetition
 - BMLNet::BMLHead, 46
- requestHeaders
 - AssetPackage::RequestSettings, 87
- RequestResponse
 - AssetPackage::RequestResponse, 84
- RequestSettings
 - AssetPackage::RequestSettings, 86
- responsMessage
 - AssetPackage::RequestResponse, 85
- responseCode
 - AssetPackage::RequestResponse, 85
- responseHeaders
 - AssetPackage::RequestResponse, 85
- ResultAllowed
 - AssetPackage::RequestResponse, 85
- Revision
 - AssetManagerPackage::RageVersionInfo, 83
- Save
 - AssetPackage::IDataStorage, 69
- SaveDefaultSettings
 - AssetPackage::BaseAsset, 20
 - AssetPackage::IDefaultSettings, 72
- SaveSettings
 - AssetPackage::BaseAsset, 21
- SaveVersionInfo
 - AssetManagerPackage::RageVersionInfo, 82
- ScheduledBlocks
 - AssetPackage::RageBMLNet, 79
- Settings
 - AssetPackage::BaseAsset, 23
 - AssetPackage::IAsset, 67
 - AssetPackage::RageBMLNet, 79
- SettingsFromXml
 - AssetPackage::BaseAsset, 21
- SettingsToXml
 - AssetPackage::BaseAsset, 21
- Severity
 - AssetPackage, 11
- Side
 - BMLNet::BMLFaceFacs, 35
- side
 - BMLNet::BMLFaceFacs, 35
- Success
 - AssetPackage::IWebServiceResponseAsync, 76
- SyncPointCompleted
 - AssetPackage::RageBMLNet, 78
- syncPoints
 - BMLNet::BMLBlock, 30
- target
 - BMLNet::BMLGaze, 40
 - BMLNet::BMLHeadDirectionShift, 47
 - BMLNet::BMLLocomotion, 49
 - BMLNet::BMLPointing, 51
- text
 - BMLNet::BMLSpeech, 58
- Timer
 - AssetPackage::RageBMLNet, 80
- TriggerSyncPoint
 - AssetPackage::RageBMLNet, 79
 - BMLNet::BMLSyncPoint, 61
- TryParseAttribute< T >
 - BMLNet::BMLBlock, 28
- TryParseSyncPoint
 - BMLNet::BMLBlock, 29
- Type
 - BMLNet::BMLStance, 59
- type
 - BMLNet::BMLStance, 60
- Update
 - AssetPackage::RageBMLNet, 79
 - BMLNet::BMLSyncPoint, 61
- UpdateDefaultValues
 - AssetPackage::BaseSettings, 24
- uri
 - AssetPackage::RequestSettings, 87
- Version
 - AssetPackage::BaseAsset, 23
 - AssetPackage::IAsset, 67
- VersionAndDependenciesReport
 - AssetManagerPackage::AssetManager, 16
- VersionInfo
 - AssetPackage::BaseAsset, 23
- WebServiceRequest

AssetPackage::IWebServiceRequest, [74](#)
WebServiceRequestAsync
AssetPackage::IWebServiceRequestAsync, [75](#)