FEATURES

Sign,milepost or similar object in Linear System.

Table

Main object

psdi.app.feature.FeatureSet

UniqueID: FEATURESID

Primary key: FEATURE

LOGICAL RELATIONSHIPS

FOREIGN KEYS INTO FEATURES

Object(Parent Keys)	Target Object(Target Keys)	Rel Number	Description
FEATURES(FEATURE)	ASSETFEASPECHIST(ENDASSETFEATURE)	1 to many	End Asset Feature
FEATURES(FEATURE)	ASSETFEASPECHIST(FEATURE)	1 to many	Feature
FEATURES(FEATURE)	ASSETFEASPECHIST(STARTASSETFEATURE)	1 to many	Start Asset Feature
FEATURES(FEATURE)	ASSETFEATURE(ENDFEATURE)	1 to many	End Asset Feature
FEATURES(FEATURE)	ASSETFEATURE(FEATURE)	1 to many	Asset Feature
FEATURES(FEATURE)	ASSETFEATURE(STARTFEATURE)	1 to many	Start Asset Feature
FEATURES(FEATURE)	ASSETFEATUREHIST(ENDFEATURE)	1 to many	End Feature
FEATURES(FEATURE)	ASSETFEATUREHIST(FEATURE)	1 to many	Feature
FEATURES(FEATURE)	ASSETFEATUREHIST(STARTFEATURE)	1 to many	Start Feature
FEATURES(FEATURE)	ASSETFEATURESPEC(ENDFEATURE)	1 to many	End Feature
FEATURES(FEATURE)	ASSETFEATURESPEC(FEATURE)	1 to many	Feature
FEATURES(FEATURE)	ASSETFEATURESPEC(STARTFEATURE)	1 to many	Start Feature
FEATURES(FEATURE)	ASSETMETER(ENDFEATURE)	1 to many	End Feature
FEATURES(FEATURE)	ASSETMETER(STARTFEATURE)	1 to many	Start Feature
FEATURES(FEATURE)	ASSETSPECHIST(ENDASSETFEATURE)	1 to many	End Asset Feature
FEATURES(FEATURE)	ASSETSPECHIST(STARTASSETFEATURE)	1 to many	Start Asset Feature
FEATURES(FEATURE)	FEATURESPEC(FEATURE)	1 to many	Feature
FEATURES(FEATURE)	FEATURESTATUS(FEATURE)	1 to many	Status History
FEATURES(FEATURE)	MULTIASSETLOCCI(FEATURE)	0 to 1	Features
FEATURES(FEATURE)	MULTIASSETLOCCIPR(FEATURE)	1 to many	Feature

FEATURES FOREIGN KEYS

Object(Parent Keys)	Target Object(Target Keys)	Rel Number	Description
CLASSSTRUCTURE(CLASSSTRUCTUREID)	FEATURES(CLASSSTRUCTUREID)	1 to many	Class Structure
LANGUAGE(MAXLANGCODE)	FEATURES(LANGCODE)	1 to many	Language for the record

COLUMNS

Attribute	Modifier	Title	Remarks	SameAsAttrib	SameAsObject
FEATURESID	Required	Unique Id	Identifies the Features record. This value must be unique for all Features records.		
FEATURE	Required	Feature	An object that exists on or alongside a linear asset that is not a point asset (for example, milepost, guardrail)		

Attribute	Modifier	Title	Remarks	SameAsAttrib	SameAsObject
FEATURETYPE	Required	Type	There are two types of features. A POINT feature has no length, or its length is not related to its use with the linear asset. For example, a mile post and a power line crossing are both point features. A LINEAR feature has length and its length is related to the linear asset. For example, a guard rail has length and its length is described by start and end measures along the roadway.		
CONTINUOUS	Required	Continuous	Specifies whether the feature is continuous. If the check box is selected, it is continuous. A continuous feature exists for the entire span of the linear asset with which it is associated. To satisfy this condition you can create a single Features record whose start and end measures equal the start and end measures of the linear asset. You can also create multiple Features records that together cover the entire linear asset without overlapping. To validate this property, run the Gap and Overlap report. If the check box is cleared (the default), you can apply the feature to a segment of the linear asset.		
DESCRIPTION		Description	Describes the feature. To enter or view additional information, click the Long Description button.		
SHARED	Required	Shared	Specifies whether the feature will be shared by related linear assets. If the check box is selected, and you apply the feature to one linear asset, the feature will be visible to other linear assets that you specify as related to it. For example, if two linear assets share a segment, and you apply the feature to one of the assets, it will be visible to the other, as well.		
CLASSSTRUCTUREID		Classification	Unique ID of the classification structure that holds a list of attributes, for example, Size, Length and Color.	CLASSSTRUCTUREID	CLASSSTRUCTURE
LANGCODE	Required	Language Code	Language Code	MAXLANGCODE	LANGUAGE
DESCRIPTION_LONGDESCRIPTION	Nonperistent	DESCRIPTION Long description	Long description for DESCRIPTION		
HASLD	Required	Has Long Description	Boolean flag to indicate if there is any long description for this record.		

Attribute	Modifier	Title	Remarks	SameAsAttrib	SameAsObject
STATUS		Status	Similar to asset status, this allows us to manage features, and the access to them, without having to delete them from the database.	STATUS	ASSET
ISLINEARREF	Required	Is Reference Point	Specifies whether you can use the feature as a start or end point for a linear segment. If the check box is selected (the default), Maximo will create an feature instance that you can use as a start or end reference point. If the check box is cleared, you cannot.	ISLINEARREF	RELATION
RELATIONID		Relation Unique Id	Unique identifier for thie relationship (Relation Table) that generated this relational feature.	RELATIONID	RELATION

MAXIMO RELATIONSHIPS

MAXIMO OUTGOING RELATIONSHIPS

Name	Target	Remarks	Where Clause
WHEREFEATURE	ASSETFEATURE	Relationship to the AssetFeature table, used to find all AssetFeatures for this Feature, that is, it answers the question 'Where is this feature used?'. (assetfeature.feature=features.feature). The resulting set will contain zero or more objects.	feature=:feature
WHEREISFEATURENOW	ASSETFEATUREHIST	Relationship to the AssetFeatureHist table, used to find all AssetFeaturesHist records for this Feature that have not yet been removed. It answers the question 'Where is this feature currently used?'. (assetfeaturehist.feature=features.feature and assetfeaturehis.removeddate is null). The resulting set will contain zero or more objects.	feature=:feature and removeddate is null
CLASSSPEC	CLASSSPEC	Relationship to the ClassSpec table, used to find all class specification records for a given feature (classspec.classstructureid = features.classstructureid). The resulting set will contain zero or more objects.	classstructureid=:classstructureid
CLASSSTRUCTURE	CLASSSTRUCTURE	Relationship to the classstruvcture table, used to find the classstructure record for a given feature. (features.classstructureid=classstructure.classstructureid) The resulting set will contain zero or one object.	classstructureid = :classstructureid
DOCLINKS	DOCLINKS	Relationship to the DocLinks table, used to find all document records for a given Feature. The resulting set will contain zero or more objects.	ownertable = 'FEATURES' and ownerid = :featuresid
FEATURESPEC	FEATURESPEC	Relationship to the featurespec table, used to find the featurespec records for a given feature. (feature.feature=featurespec.feature and feature.classstructureid=featurespec.classstructureid) The resulting set will contain zero or more objects.	feature=:feature and classstructureid=:classstructureid
FEATURESSPEC	FEATURESPEC	Relationship to the featurespec table, used to find the featurespec records for a given feature. (features.featuresid=featurespec.refobjectid) The resulting set will contain zero or more object.	refobjectid=:featuresid
FEATURESSPECCLASS	FEATURESPEC	Relationship to the featurespec table, used to find the featurespec records for a given feature. (features.feature=featurespec.feature and features.classstructureid=featurespec.classstructureid) The resulting set will contain zero or more object.	feature= :feature and classstructureid = :classstructureid
FEATURESTATUS	FEATURESTATUS	Relationship to the FeatureStatus table. The resulting set will contain zero or more objects.	feature=:feature
FTRCHANGESTATUS	FTRCHANGESTATUS	Relationship to the non-persistent FTRChangeStatus table. (There is no where clause for non-persistent objects). The resulting set will contain zero or more objects. Note : FTRChangeStatus is a non-persistent MBO with whose help the dialog box binds with the object.	null

Name	Target	Remarks	Where Clause
STATUSDESC	SYNONYMDOMAIN	Relationship to SYNONYMDOMAIN.	domainid='FEATURESTATUS' and value=:status and :&DOMAINFILTER&_STATUS

MAXIMO INCOMING RELATIONSHIPS

Name	Source	Remarks	Where Clause	Cardinality
ENDFEATURE	ASSETFEATURE	Relationship to the Features table, used to find the EndFeature that delineates where this AssetFeature's Feature ends. EndFeature is null and zero objects are returned if this AssetFeature is a point feature. (features.feature=assetfeature.endfeature). The resulting set will contain zero or one object.	feature=:endfeature	UNDEFINED
FEATURE	ASSETFEATURE	Relationship to the Features table, used to find the feature for this AssetFeature. (features.feature=assetfeature.feature). The resulting set will contain one object.	feature=:feature	UNDEFINED
STARTFEATURE	ASSETFEATURE	Relationship to the Features table, used to find the StartFeature that delineates where this AssetFeature's Feature begins. StartFeature is null and zero objects are returned if this AssetFeature is a point feature. (features.feature=assetfeature.startfeature). The resulting set will contain zero or one object.	feature=:startfeature	UNDEFINED
FEATURE	ASSETFEATUREHIST	Relationship to the Features table, used to find the feature for this AssetFeatureHist. (features.feature=assetfeaturehist.feature). The resulting set will contain one object.	feature = :feature	UNDEFINED
FEATURE	ASSETFEATURESPEC	Relationship to the Features table, used to find the feature for this AssetFeatureSpec.	feature=:feature	UNDEFINED
FEATURES	CLASSSTRUCTURE	Relationship to the features table, used to find the features records for a given classstructure. (classstructure.classstructureid = features.classstructureid). The resulting set will contain zero or more objects.	classstructureid in (select classstructureid from classancestor where ancestor=:classstructureid)	UNDEFINED
FEATURES_ONLY	CLASSSTRUCTURE	Relationship to the features table, used to find the features records for a given classstructure. (classstructure.classstructureid =features.classstructureid). The resulting set will contain zero or more objects.	classstructureid =:classstructureid	UNDEFINED
FEATURE	PM	Relationship to the Features table, used to find the feature for this PM. (features.feature=pm.feature). The resulting set will contain one object	feature = :feature	UNDEFINED
FEATURES	RELATION	Relationship to the Features table , used to find Features record for a given relation. (Relation.relationid= Features.relationid). The resulting set will contain zero or one object.	relationid=:relationid	UNDEFINED
RELATIONFEATURES	RELATION	Relationship to the Features table , used to find Features record for a given relation. (Relation.relationnum= Features.feature). The resulting set will contain zero or one object.	feature=:relationnum	UNDEFINED
ENDFEATURE	ROUTE_STOP	Relationship to the features table, used to find the end feature on the route stop. The resulting set will contain zero or one object.	feature=:endfeature	UNDEFINED
STARTFEATURE	ROUTE_STOP	Relationship to the features table, used to find the start feature on the route stop. The resulting set will contain zero or one object.	feature=:startfeature	UNDEFINED