**Componentimplementationimpl**:

Change: basicGetType now calls on getImplemented (with proxy resolved) – originally basicGetImplemented on Realization.

Needed because an isSet method invokes basicGetType for categories such as System and the returned proxy is not system type, but the first instantiable type.

Note: getType does resolves proxy after calling on basicGetType but is not called by isSet method in EMF.

**ComponentClassifier**:

Added requiresModes Boolean attribute. It applies to all mode declarations in the classifier. Currently it is recorded in each mode object (but not mode transition object).

[Added it to Meta model, but could not regenerate the Meta model. Added the methods in by hand.

**Modes**

In types we allow xero or more mode and zero or more transition declarations, first modes then mode transitions: (m)\* (mt)\*

In implementations we allow one or more m | mt: (m | mt )+

**Subcomponents**

Refined to: subcomponent needs reference to subcomponent being refined. Name is used in reference as well as name. Need to tag subcomponent object with a Boolean refinedTo such that we can resolve the reference that is not explicitly kept.

**ComponentType**

Cannot do common subrule (as in ANTLR grammar – would need node class instance to be passed in. Cannot do the reverse, i.e., special rules returning to common rule (ComponentType) and then assign the result to package section since we currently assign each concrete type separately and derive the ownedClassifier feature.

ComponentType returns aadl2::ComponentType:

(SystemType | ProcessType)

( 'flows'

(( ownedFlowSpecification+=FlowSpec )+ | noFlows?=NONE )

)?;

SystemType **returns** *aadl2::SystemType*:

'system' name=ID

(ownedExtension=TypeExtension)?

( 'features' ( noFeatures?=NONE |

(ownedDataPort+=DataPort| ownedEventPort+=EventPort | ownedEventDataPort+=EventDataPort)+ ) )?;

**Operations**

commented out OCL-based checks.

ClassifierOperations:

NO\_CYCLES\_IN\_GENERALIZATION\_\_DIAGNOSTIC\_CHAIN\_MAP\_\_EOCL\_INV

SPECIALIZE\_TYPE\_\_DIAGNOSTIC\_CHAIN\_MAP\_\_EOCL\_INV

ElementOperations

HAS\_OWNER\_\_DIAGNOSTIC\_CHAIN\_MAP\_\_EOCL\_INV

NamedElementOperations

HAS\_QUALIFIED\_NAME\_\_DIAGNOSTIC\_CHAIN\_MAP\_\_EOCL\_INV

NamespaceOperations

MEMBERS\_DISTINGUISHABLE\_\_DIAGNOSTIC\_CHAIN\_MAP\_\_EOCL\_INV

**XTend Check**

Works ok in domainmodel example but cannot get it to work for aadl2. May have to do with importing an existing ecore model. Neither import works.

//import aadl2;

**import** "platform:/resource/edu.cmu.sei.aadl.aadl2/model/aadl2.ecore" ;

**context** DataType **WARNING** "Name should have true" :

**false**;