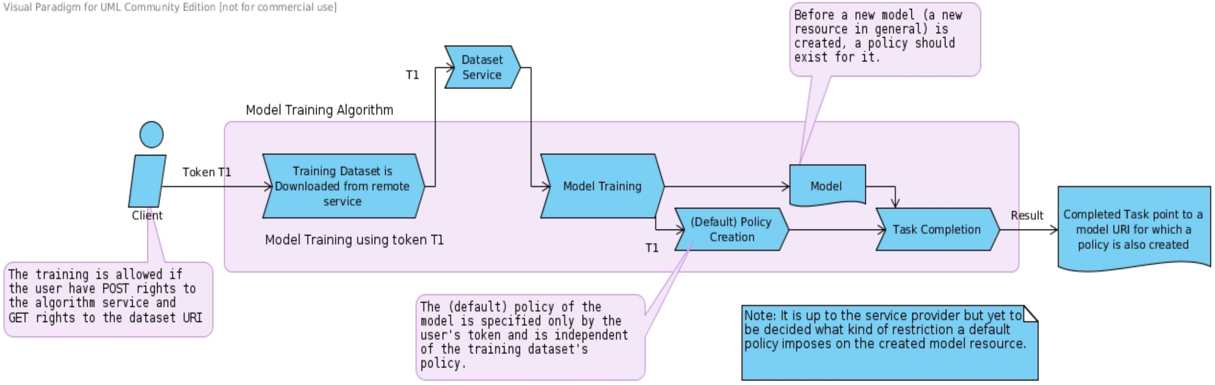
Dear All,

Yesterday we had a discussion with Nina, Tobias and Luchesar regarding some issues with authentication and authorization that need to be solved before proceeding to further implementation. Till now it has been possible to have authentication of a client using tokens but it remains ambiguous who creates the policies and how. So summarizing yesterday's discussion, we have to confront with the following use case scenarios:

1. **Override default policies:** When a client creates a resource should be able to specify a policy for it passing some parameters to the corresponding service. This can be done for simplicity using POST parameters like "policy=public" or "allow_users_get=john,nick", "allow_users_post=nick" or "allow_groups_get=development,partner" etc. Of course the client should be able to specify a policy providing an XML document for it. To avoid passing the policy as a form parameter (in application/x-form-urlencoded) a Header parameter can be used instead (e.g. Policy = "Policy: <XML for policy>".
2. **Default Policy for Model Training:** This is a relatively simple case since model creation does not reveal any information about the training dataset apart from the URI of it. The default policy is not therefore affected by the policy of the training dataset and is solely specified by the algorithm service. This is probably to be specified by the service provider though it would be better to have an agreement on what privileges an algorithm should provide and to whom.

   
   Dataset Service
T1
Model Training Algorithm
Token T1
Client
Training Dataset is Downloaded from remote service
Model Training using token T1
Model Training
Model
(Default) Policy Creation
T1
Task Completion
Result
Completed Task point to a model URI for which a policy is also created

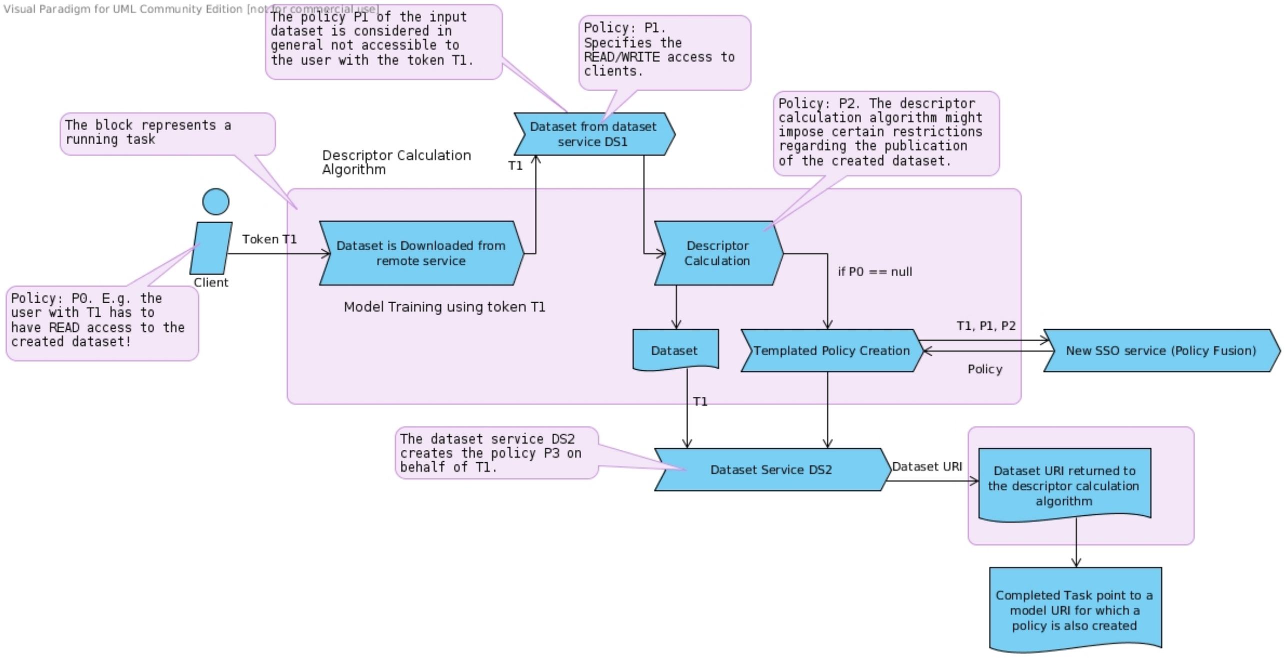
   Before a new model (a new resource in general) is created, a policy should exist for it.

   The training is allowed if the user have POST rights to the algorithm service and GET rights to the dataset URI

   The (default) policy of the model is specified only by the user's token and is independent of the training dataset's policy.

   Note: It is up to the service provider but yet to be decided what kind of restriction a default policy imposes on the created model resource.

   A reasonable default policy for model creation, can be that only the creator can
3. **Default Policy for Prediction and Descriptor Calculation:** A user with certain privileges defined through its policies (on his own resources) and other users' policies, creates a model using a dataset to which it has access (passing his token to the dataset service). The algorithm itself has also a policy that might allow the user to invoke the training service but implies nothing about whether the created dataset (containing calculated descriptors) is allowed to be published [Cases of MOPAC and 'Dragon Descriptors' algorithms]. So there is an ambiguity here regarding the policy of the published dataset (Which individuals and/or groups have access privileges). Using the 'Policy' Header parameter, the algorithm service can pass a policy XML to the dataset service and then the latter decides the final policy for it taking into account it's own restrictions and the restrictions imposed by that policy. Note that this policy, passed from one service to the other is not a policy that is directly posted to the SSO policy service and that the URI of the resource that will be created is not yet know to the intermediate services.


Visual Paradigm for UML Community Edition [not for commercial use]
The policy P1 of the input dataset is considered in general not accessible to the user with the token T1.
Policy: P1. Specifies the READ/WRITE access to clients.
Policy: P2. The descriptor calculation algorithm might impose certain restrictions regarding the publication of the created dataset.
The block represents a running task
Descriptor Calculation Algorithm
Dataset from dataset service DS1
T1
Token T1
Client
Dataset is Downloaded from remote service
Descriptor Calculation
if P0 == null
Policy: P0. E.g. the user with T1 has to have READ access to the created dataset!
Model Training using token T1
Dataset
Templated Policy Creation
T1, P1, P2
New SSO service (Policy Fusion)
Policy
T1
The dataset service DS2 creates the policy P3 on behalf of T1.
Dataset Service DS2
Dataset URI
Dataset URI returned to the descriptor calculation algorithm
Completed Task point to a model URI for which a policy is also created

Note that the client can always override the default policy and specify a custom one. That would be passed to the service that is supposed to create the resource (Hint: if it's not you the policy creator pass it to all other services you invoke).

4. **Publicly available data:** Data that are publicly available were considered till now to belong to a user called 'guest' and has the least access privileges to the overall system. However this makes it difficult to inherit these privileges to other users and especially not guests. So we suggest the creation of a universal group to which all users (including any kind of guests or anonymous users) belong. A resource will be said to be publicly available if it provides access to that group. Also, policies created for publicly available data should not allow PUT (modification of resources) nor DELETE and there will be no warranty for their persistence.

It's not yet clear how a resource can be made available to the intersection of two or more groups. If one creates a set of different policies for the same resource with different policy subjects, does the overall policy suggests that the resource is available only to the individuals in the intersection of these subjects?

The bad news is that Andreas has to implement a new web service allowing for the creation of template policies accepting as input a set of policies and an authentication token. The good news however is that we found a fancy name for it: **Policy Fusion Service!** So, you have a motivation ;-)

We should starting working on these issues, discuss them on the mailing list and take some decisions on Monday's meeting in order to proceed with the implementation of A&A and include some protected data in OpenTox too.

Best Regards,
Pantelis