

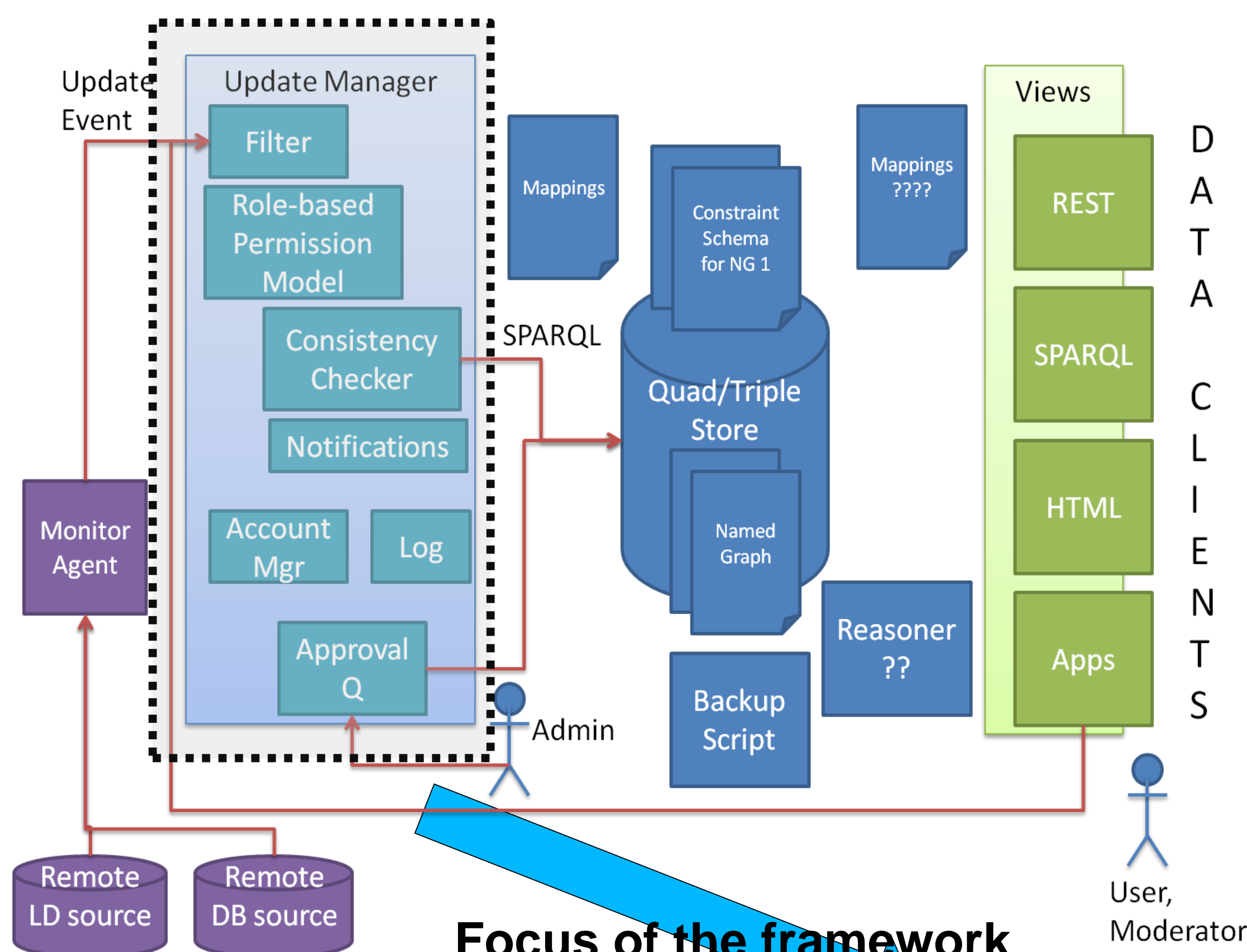
Manageable Dataset Curation for Linked Data

Wei Tai, Kevin Feeney, Rob Brennan, Declan O'Sullivan
Knowledge & Data Engineering Group (KDEG) & FAME,
School of Computer Science & Statistics, Trinity College Dublin, Dublin, Ireland.
{ Wtai, Kevin.Feeney, Rob.Brennan, Declan.OSullivan} @scss.tcd.ie
<http://kdeg.cs.tcd.ie>

Motivation and Goal:

- The success of linked data has highlighted the challenge of effectively curating the data and associated schemata or ontologies: published data tends to be 'dirty' due to its open characteristic causing maintenance overhead.
- The final goal of this work is to produce software that will help us to curate publicly available semantically-rich linked-data sets in such a way that these data-sets will remain useful over time and will require minimal maintenance effort.

Framework overview



Framework overview:

- Update manager**
 - Update control
 - The focus of this entire framework
- Data store**
 - Stores/Updates linked dataset through SPARQL
 - 4 Store
- Linked data viewing/publishing interfaces**
 - Third party interfaces

Update types

Insert triple	Delete triple
Insert class	Delete class
Modify label	Modify status
Insert property	Delete property
Modify domain	Modify range
Delete domain	Insert subclassof
Insert subpropertyof	Delete subclassof
Delete subpropertyof	Insert instance
Insert property link	Delete instance
Delete property link	

Example system/semantic constraints

Updates must contain no anonymous node.

A term must have a label

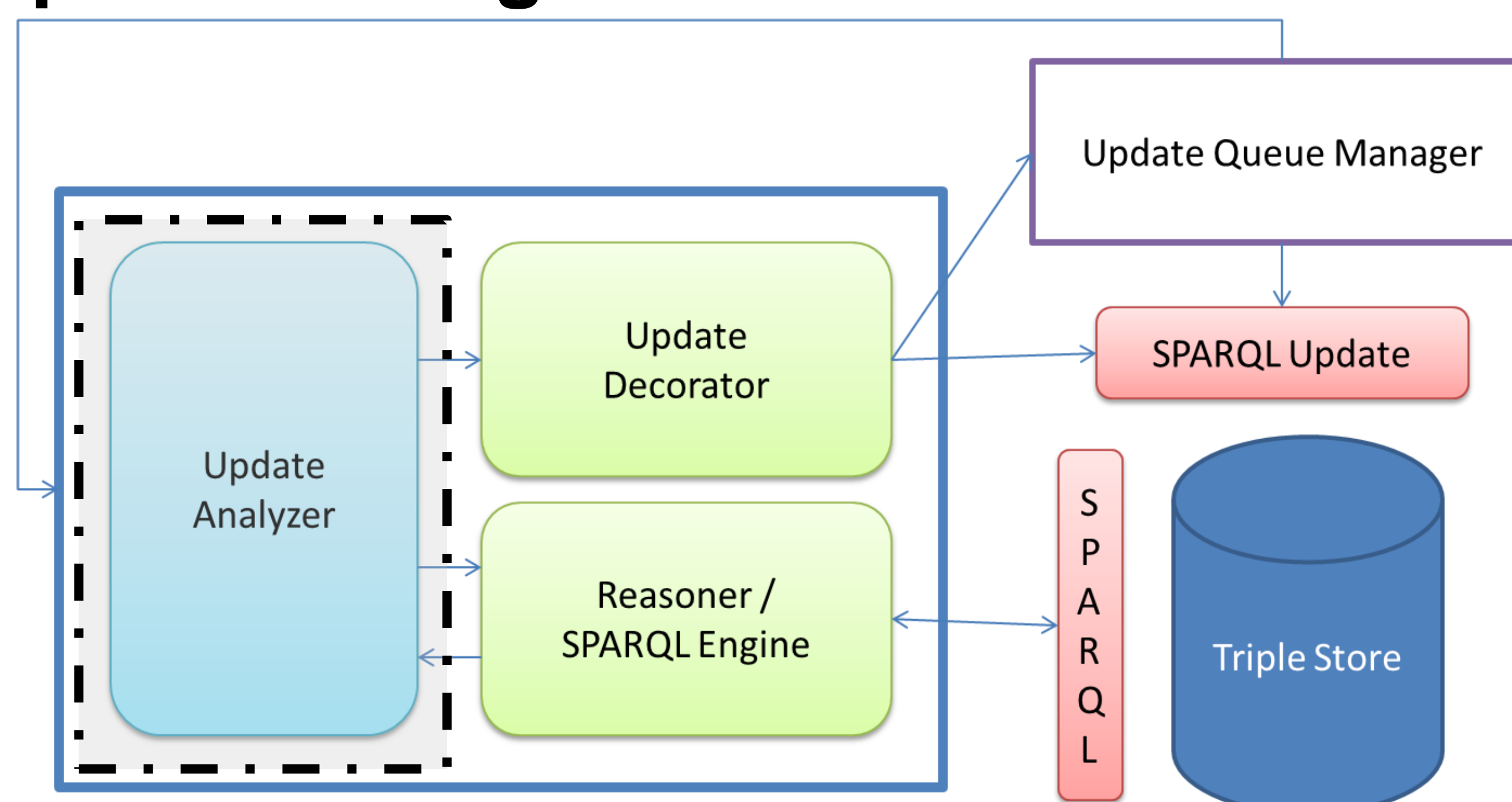
A user-defined individual must be an type of user-defined class.

A class to be deleted must not have instances

A property to be deleted must not be used to link instances

.....

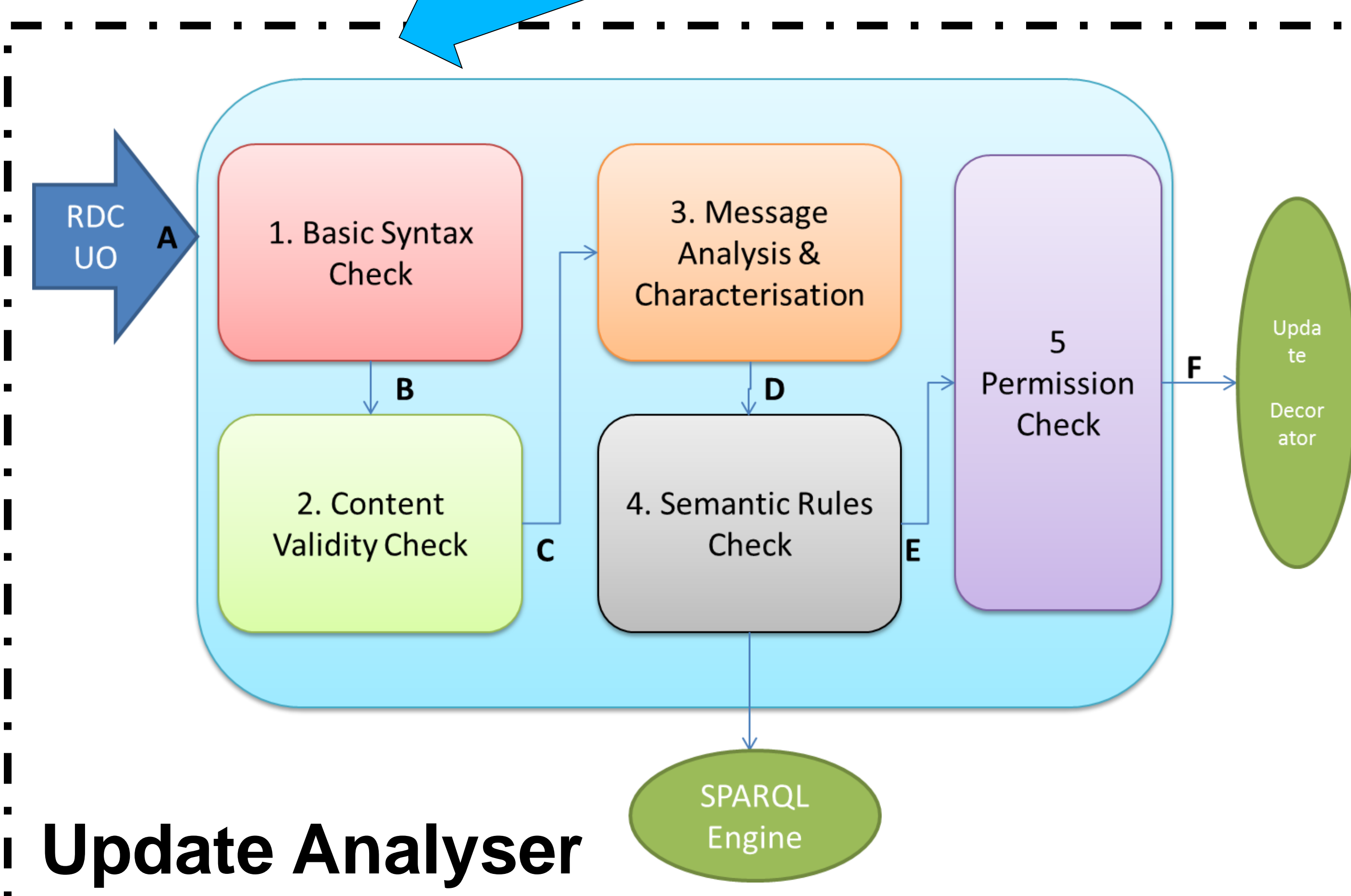
Update manager



Update manager:

- Update analyser**
 - Update validity check
 - Semantic check
 - Permission check
- Update decorator**
 - Generates SPARQL update
- Update queue manager**
 - Holds the updates need for moderator's review

Where all the updates are checked



Update Analyser

Update analyser:

- Basic syntax check**
 - Checks if crude update objects (JSON) are well-formed.
- Content validity check**
 - Checks for the validity of crude update objects.
- Message analysis&characterisation**
 - Generates typed update objects from crude update objects.
- Semantic rules check**
 - Checks typed update objects against configurable system/semantic constraints and fail to comply leads to warnings or errors.
- Permission check**
 - Checks if the typed update is issued by a authorized user.