

A proposal for a Triple Space kernel architecture



Robert Tolksdorf, Elena Paslaru, **Lyndon J B Nixon**
FU Berlin
27.06.06

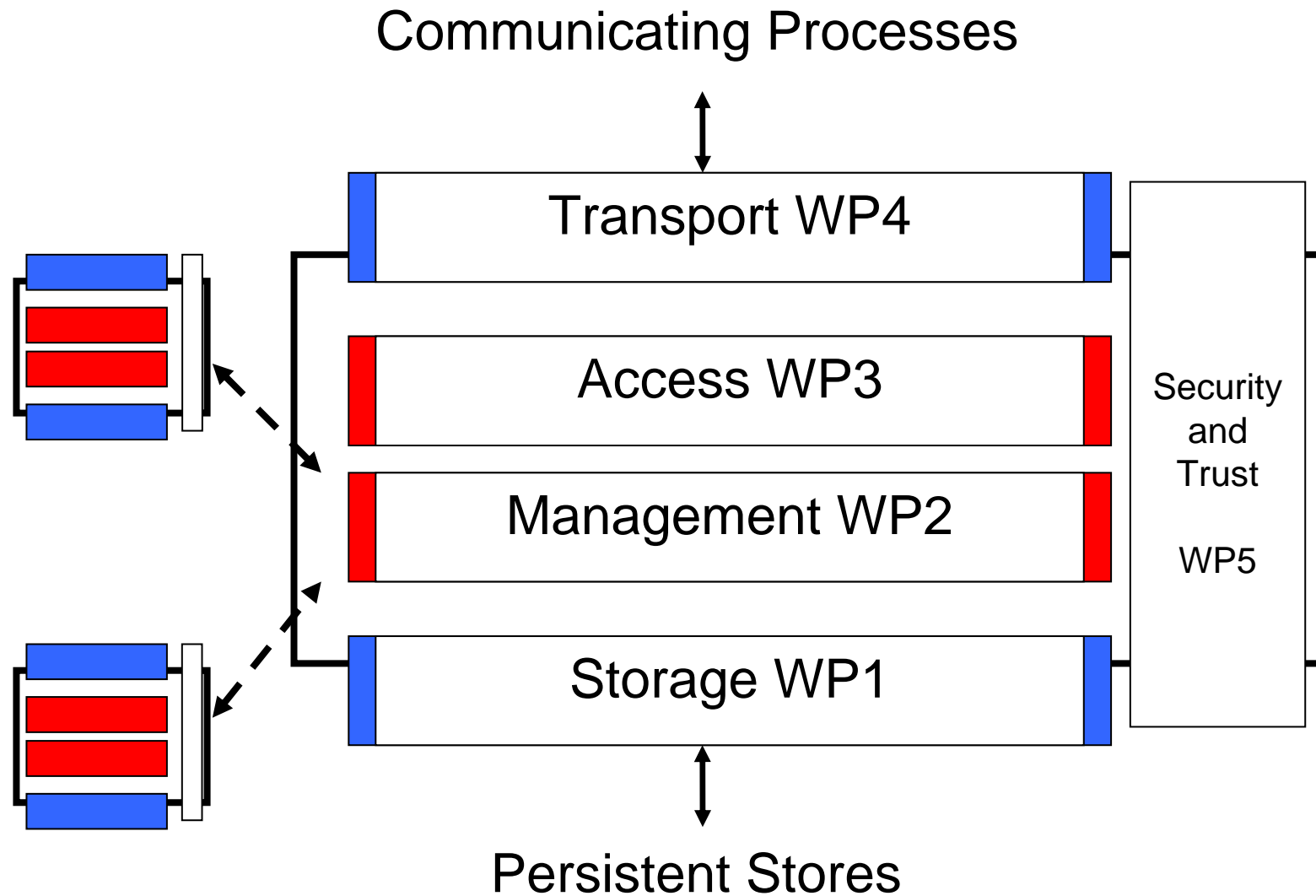


- Main objectives
- Conceptual architecture of a TripleSpace kernel
- Relationship to other WPs
- Action points

- Form a common understanding of internal structure of triplespace kernel
- Agree on the distribution of responsibilities among and within work packages
- Begin a project-wide discussion on the overall direction of development

Conceptual architecture

Triple Space layers

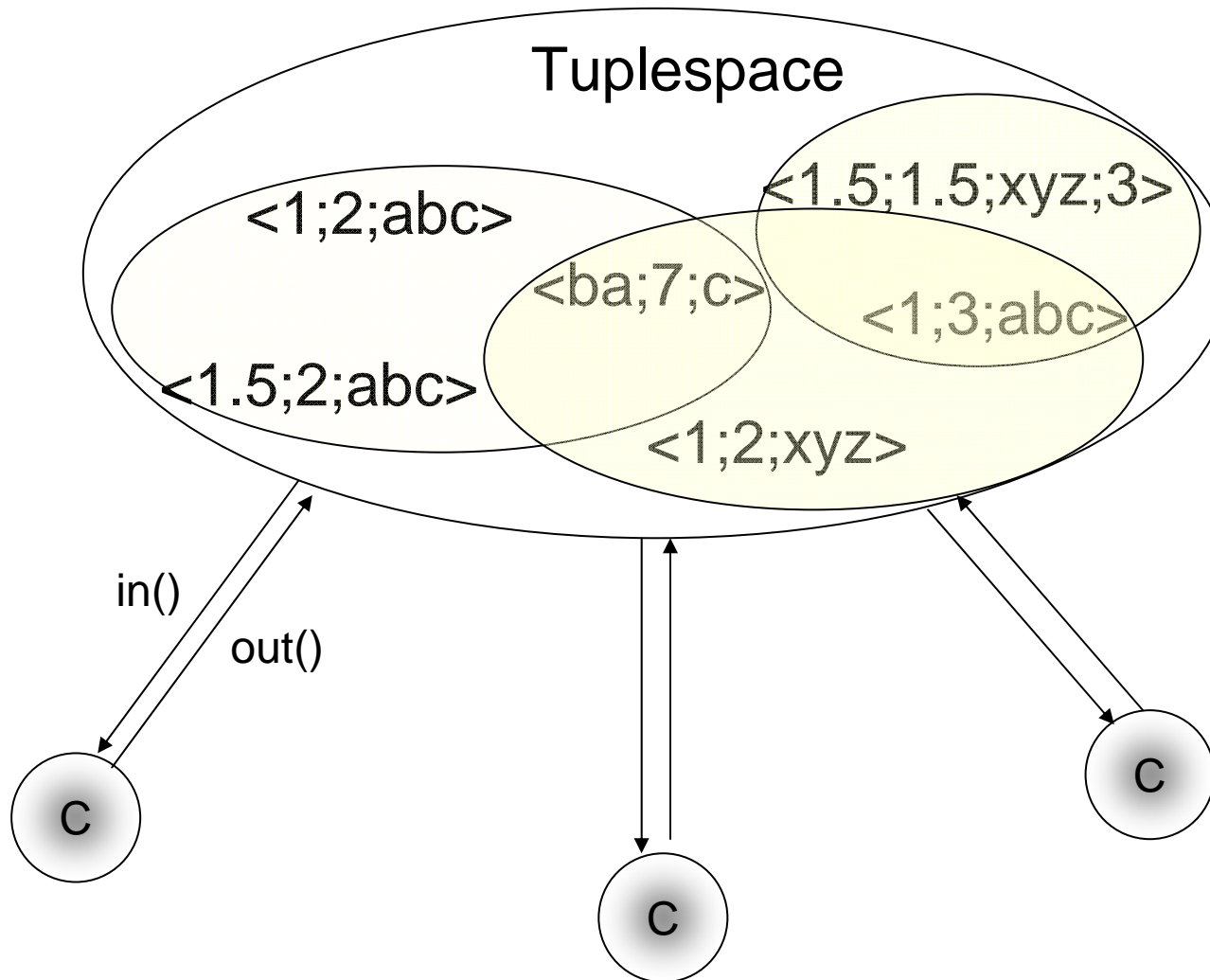


Access and management layers

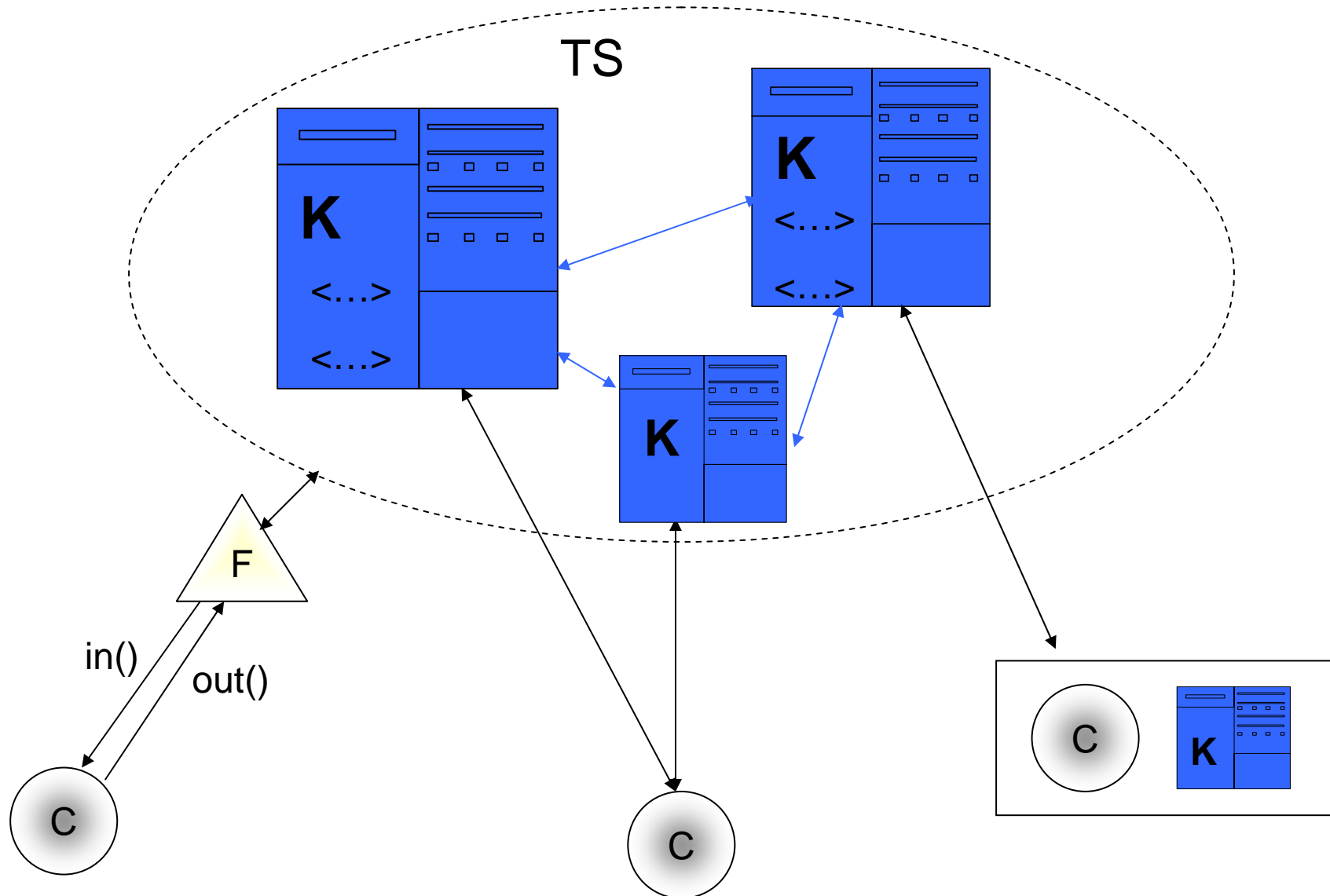


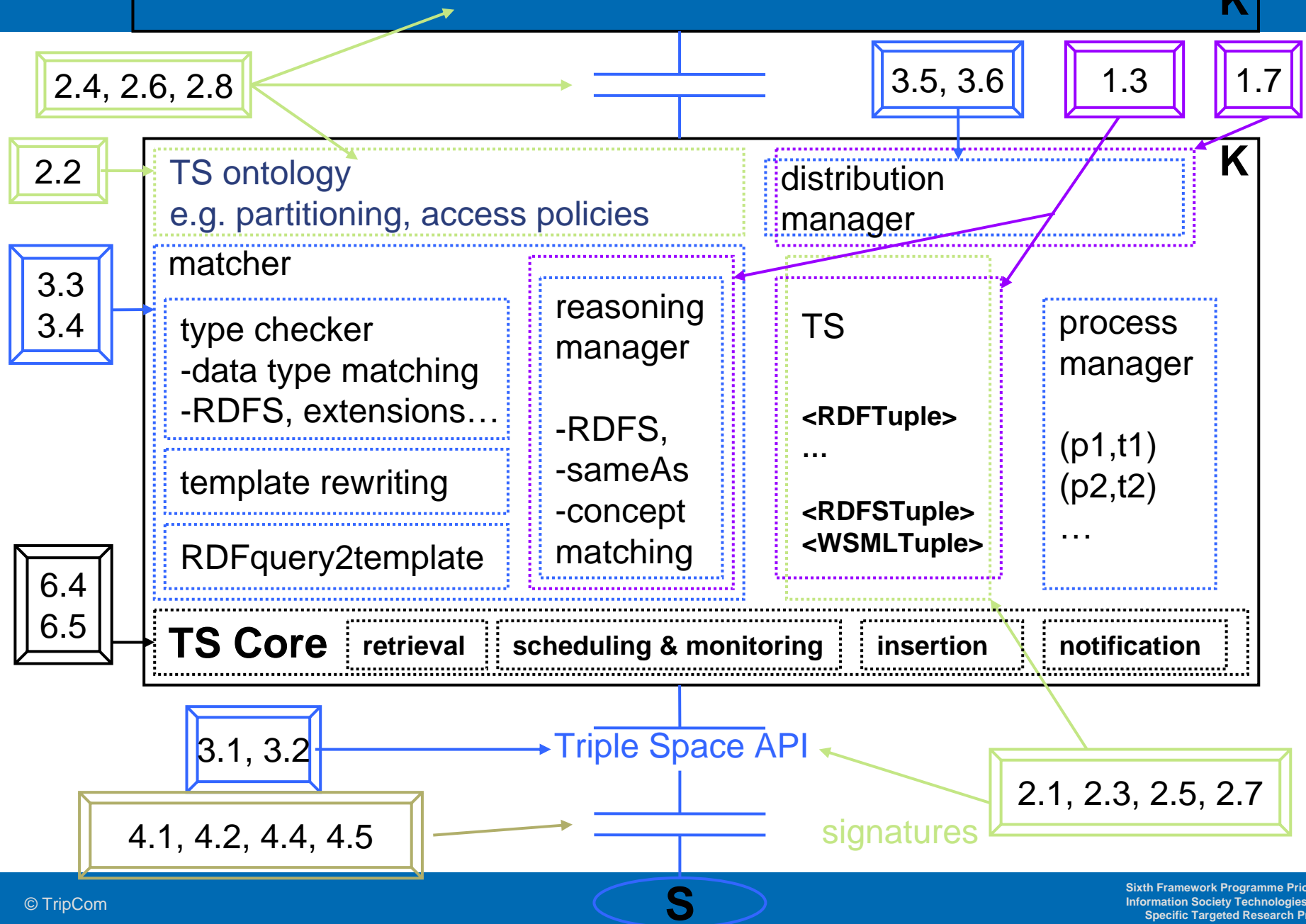
Component	Role	Functionality
Tuple/Tuplespace Management	To maintain the shared tuplespace and manage the tuples which exist in it	<ul style="list-style-type: none">▪ tuple format▪ tuple type▪ tuple organisation within a single process▪ tuple distribution across processes▪ access policies
Tuplespace Access	To support a standardized set of coordination primitives which enable clients to perform operations on the tuplespace	<ul style="list-style-type: none">▪ coordination primitives▪ parameter handling▪ input/output of tuples from/to clients

Logical view of Triple Space

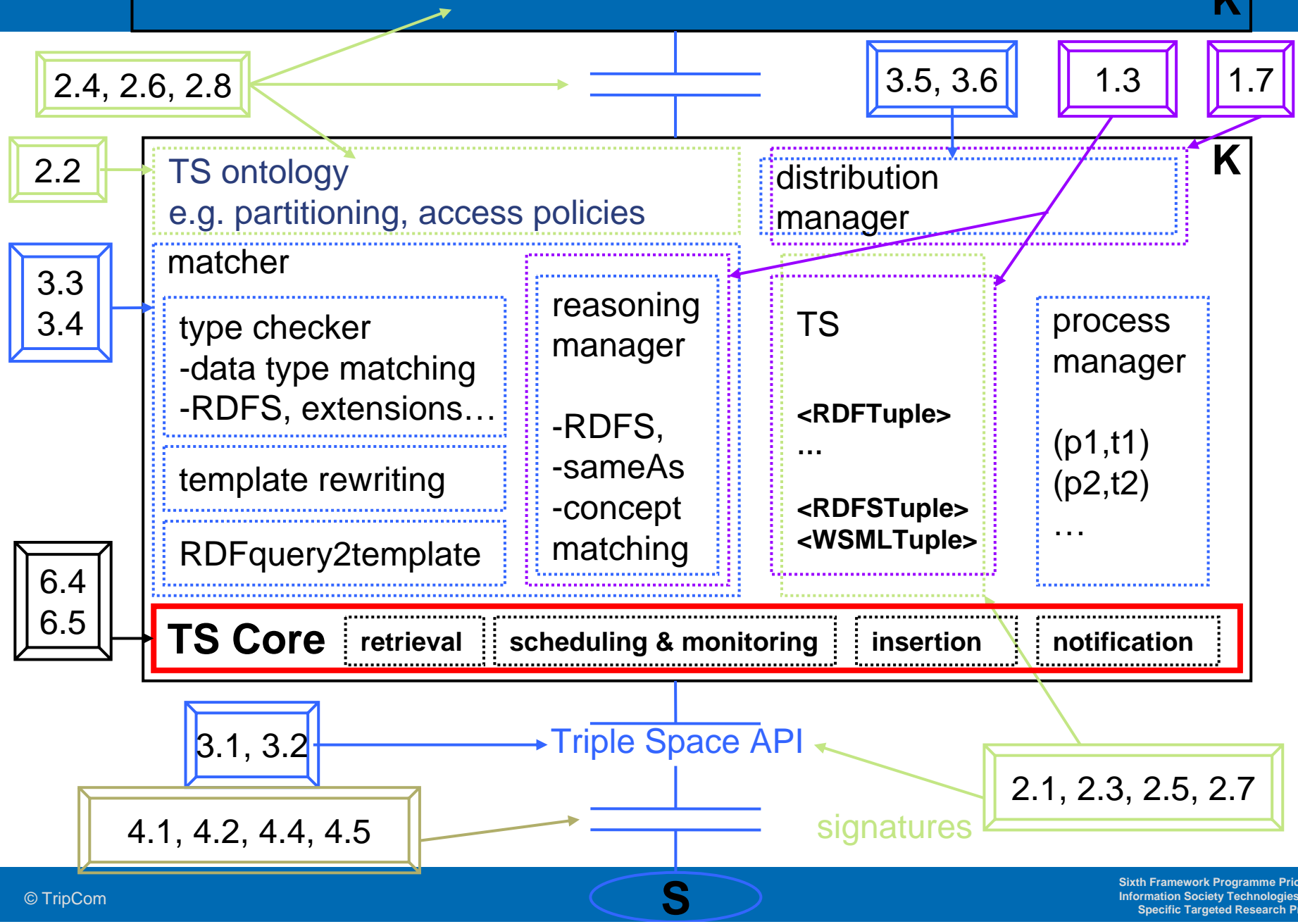


Machine view of Triple Space



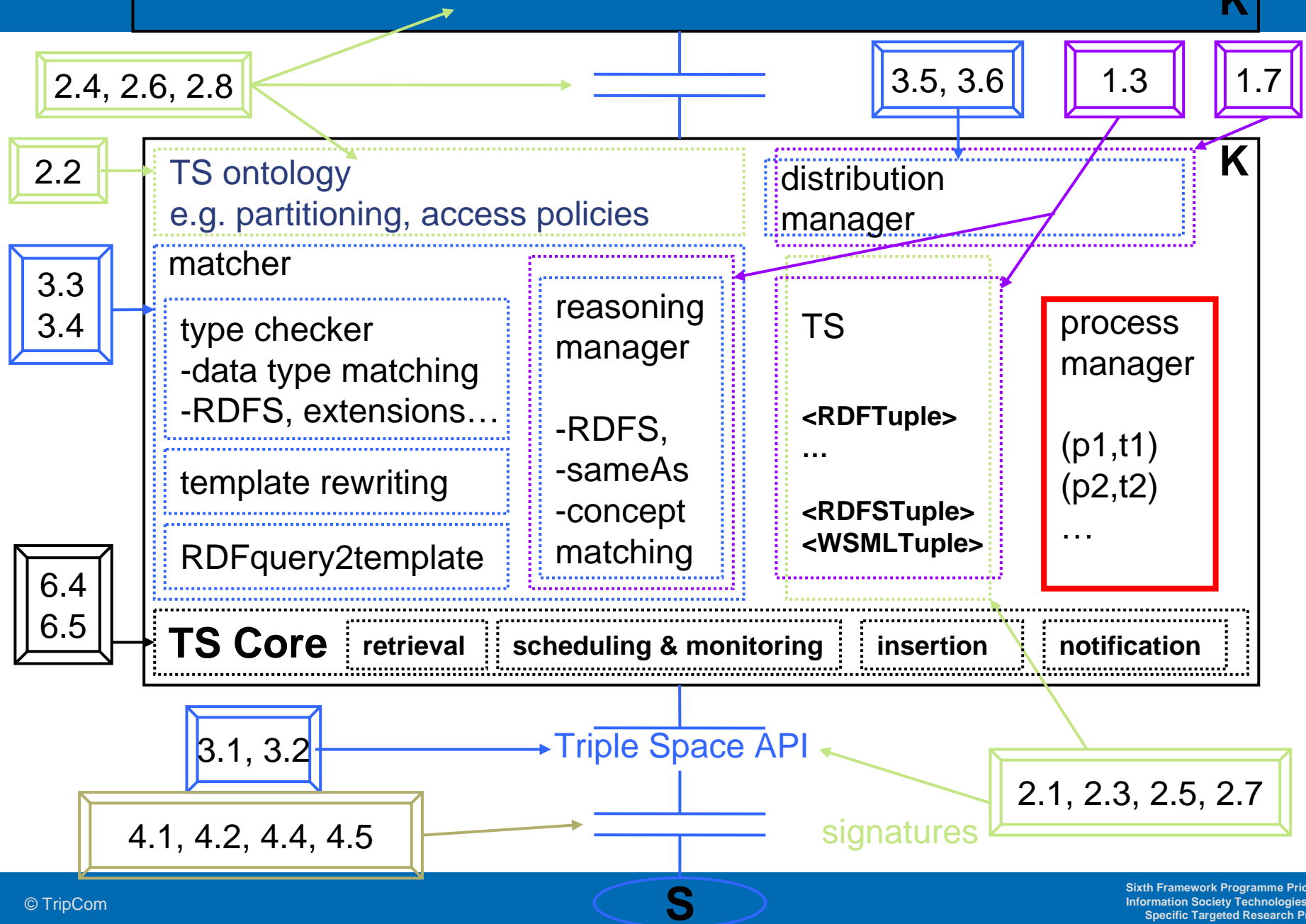


- TripleSpace requires a coordination model and a language which defines the model
- We plan to take the tuplespace paradigm and the associated Linda coordination language for these purposes
- Which extensions and revisions are required for TripCom?
- How will it map onto Web Service interaction?

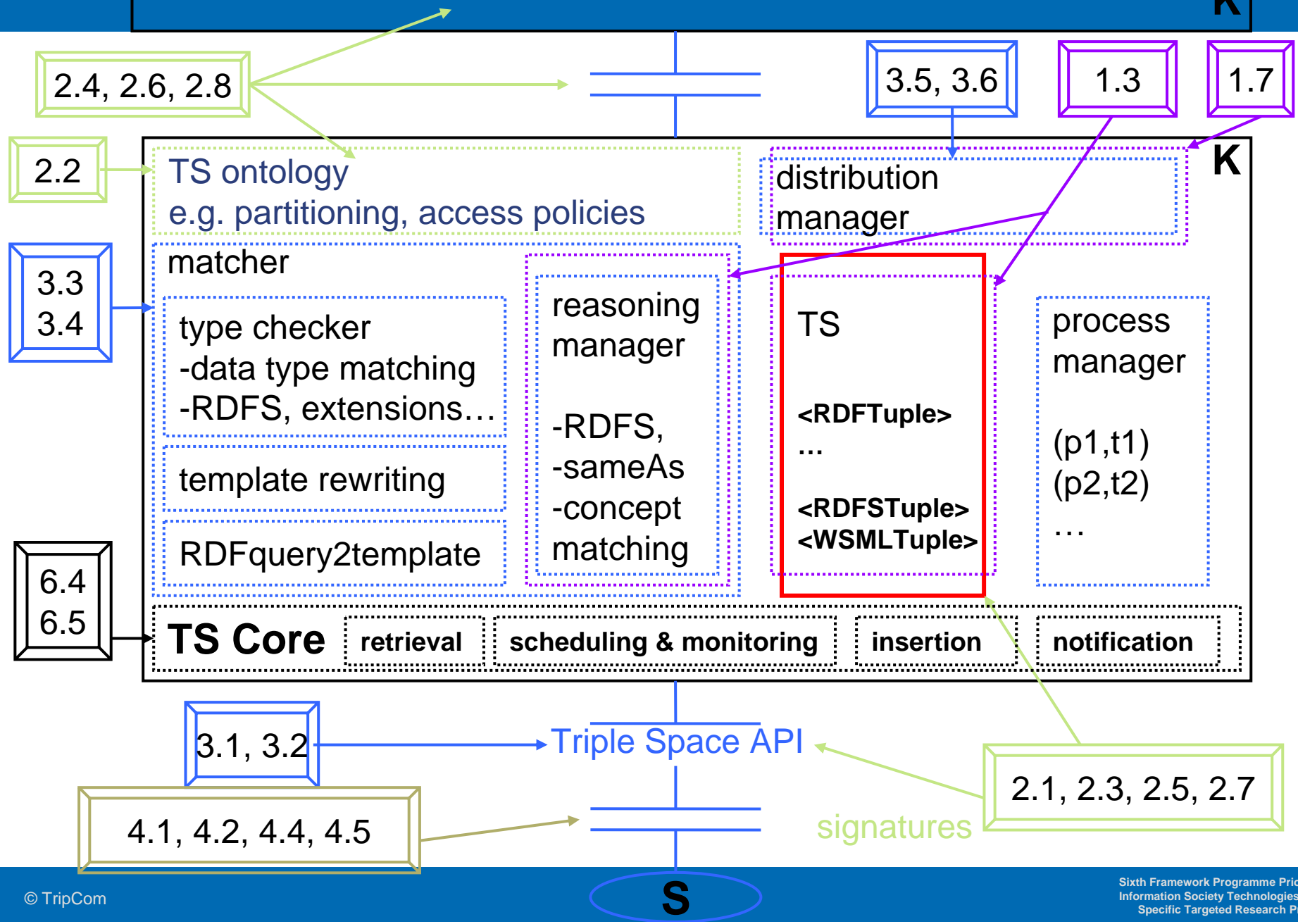


- Handles incoming operations
- Hands off tasks to appropriate components in the TripleSpace kernel

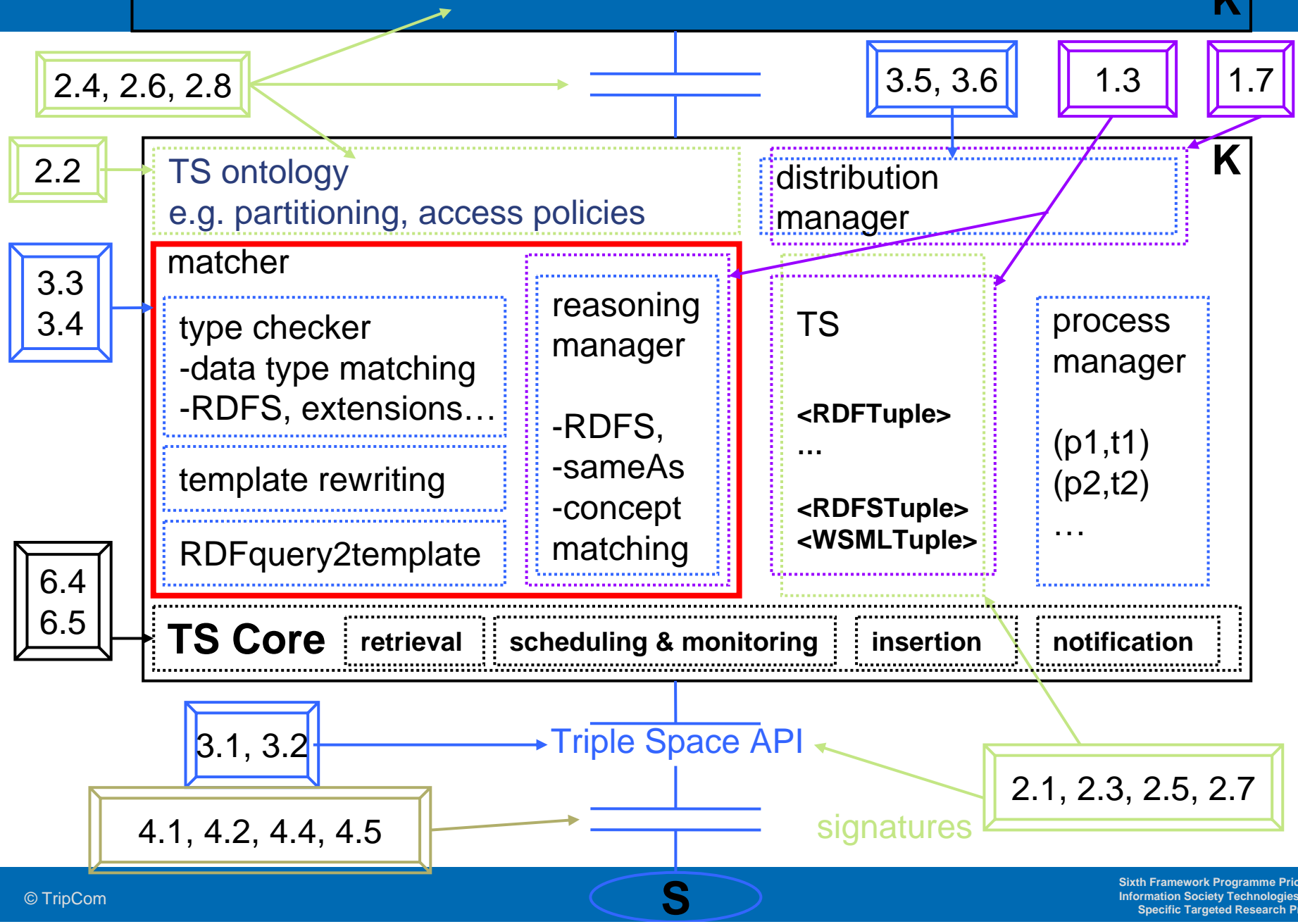
- Core functionality:
 - Insertion
 - Retrieval
 - Scheduling and monitoring
 - Notification
 - ...



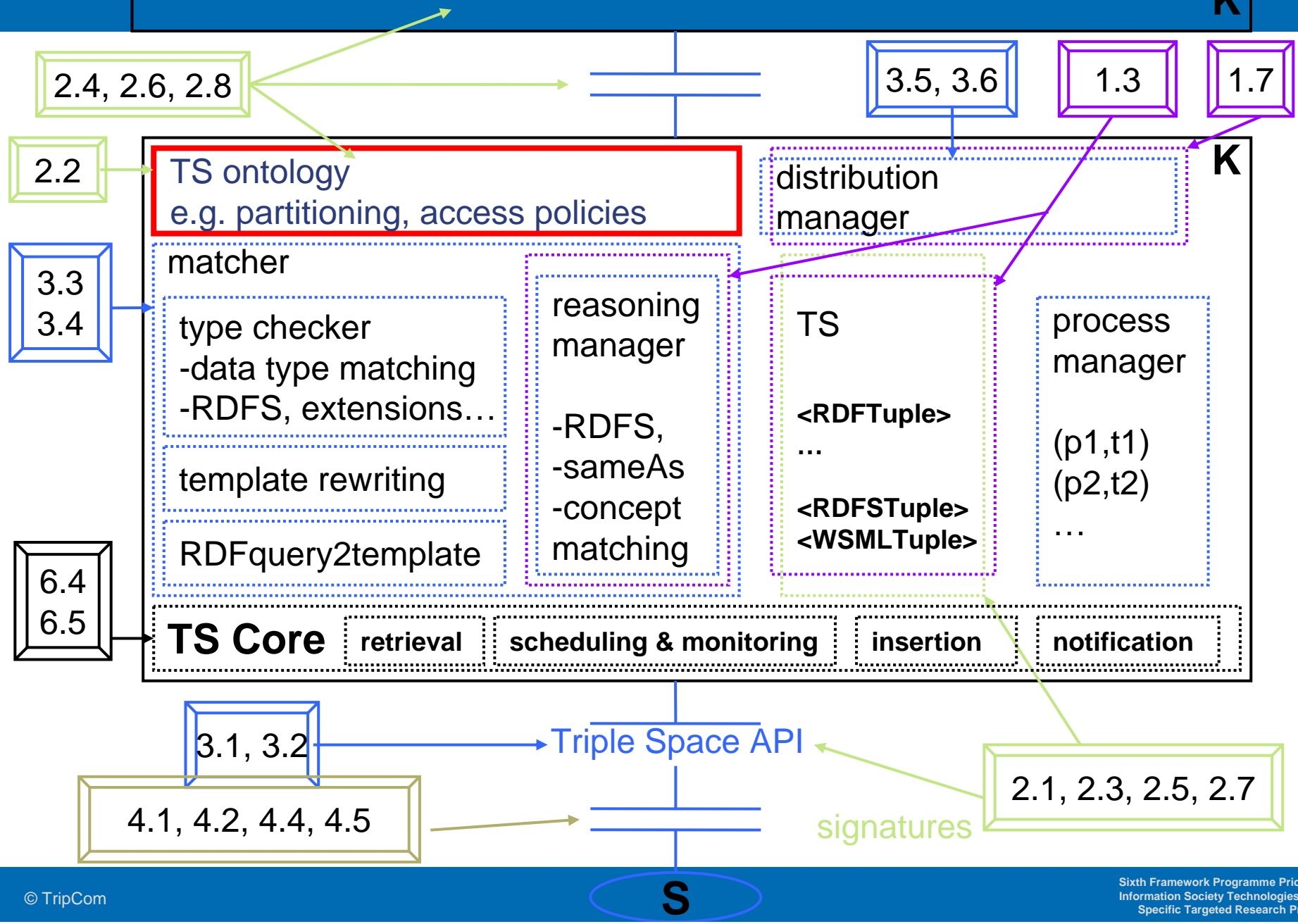
- Maintains tables for outstanding retrieval operations (blocking)
- Timeouts
- Notification mechanism



- Virtual shared space containing tuples
 - Data represented as tuples
 - Service represented as (reactive) tuples
 - Mapping tuples
 - Administrative data according to the tuplespace ontology
- The space can be further structured according to
 - Access policies
 - Tuple types
 - Agent-defined views
 - ...

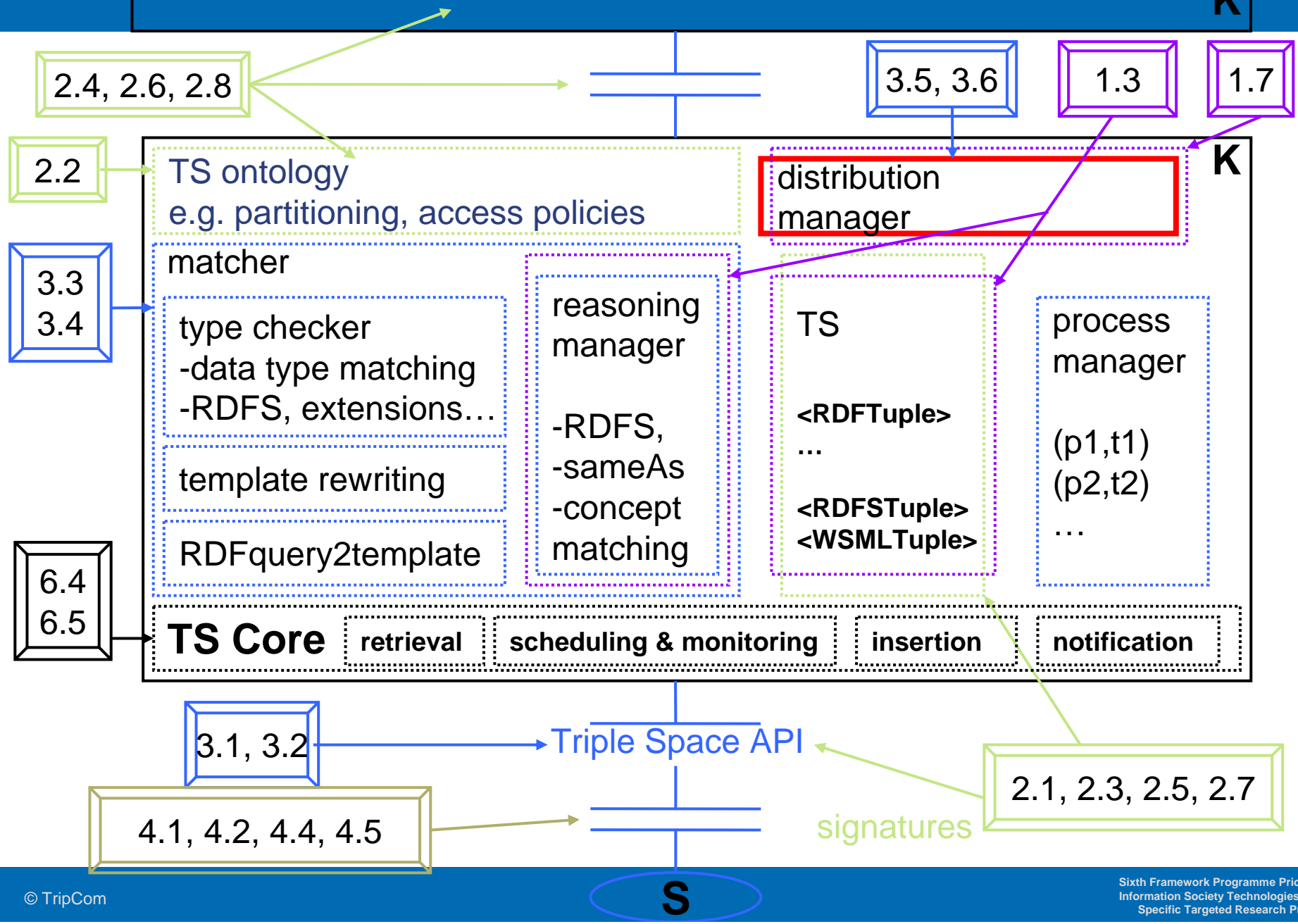


- Tuple matching must be extended to handle RDF and semantic queries
- Reasoning on application data can be handled in the storage layer
- Reasoning on administrative data is handled by the tuplespace kernel on the basis of the TS ontology
- TS API may use templates, while the storage layer will require (SPARQL) queries
- TS API could also provide support for SPARQL queries
 - When is an intermediary mapping to template necessary?



- Means to structure administrative data about the tuplespace
- Provides a basis for optimization

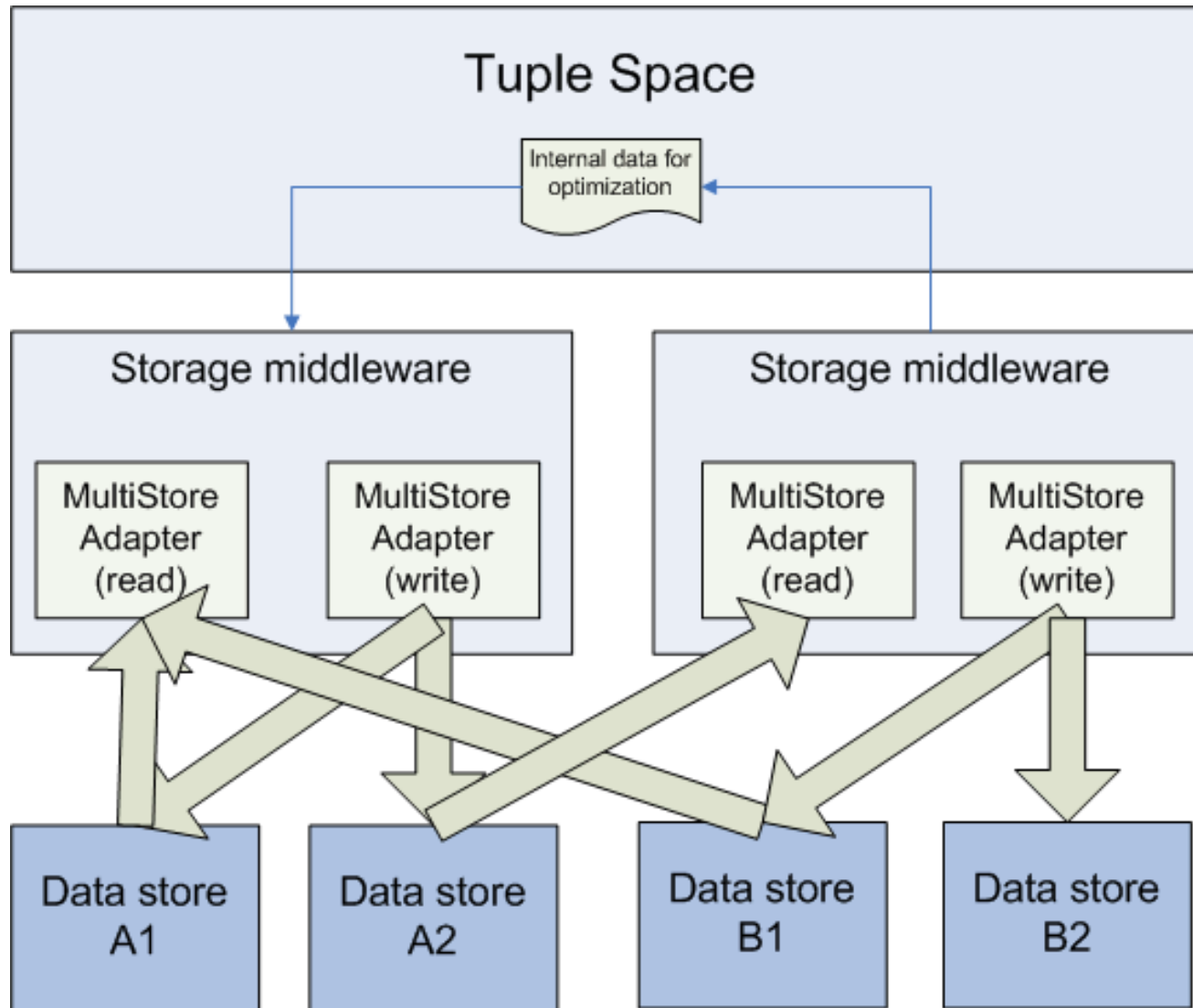
- Administrative data could include:
 - Types of tuples and tuplespaces
 - Agents and access policies
 - Operation history, logging
 - Distribution strategy



- Operation distribution according to some distribution function
- API for inter-kernel communication
- Which is optimal relationship between the distribution manager within the tuplespace kernel and the storage layer?

Relationship to other WPs

Relationship to Storage (WP1)



- To be specified
 - Mapping of the WS-communication into tuplespace
 - Mapping of the WS-registry into the tuplespace
 - Data, message and protocol mediation

- Can we specify Triple Space independently of the processes that will use it?

- Do SWS raise any requirements that must be explicitly considered in the architecture?

- Security and trust measures include:
 - Agent authentication
 - Access policy
 - Data encryption
 - Reputation
- Which components need to be extended?
- Do any aspects of the architecture require modification?

Action points

- What concrete form should the architecture take (component-oriented, service-oriented...)?
- Are the presented components sufficient for the realization of the project goals?
- How do the activities of others WPs impact on the presented architecture?

So how do we bring Triple Space to its full potential???