Software Technology Research Laboratory,

Remote Entrusting of Mobile Multi-Agent Systems

Kevin Jones

kij@dmu.ac.uk

http://www.cse.dmu.ac.uk/~kij

Re-Trust'08 - Trento, Italy

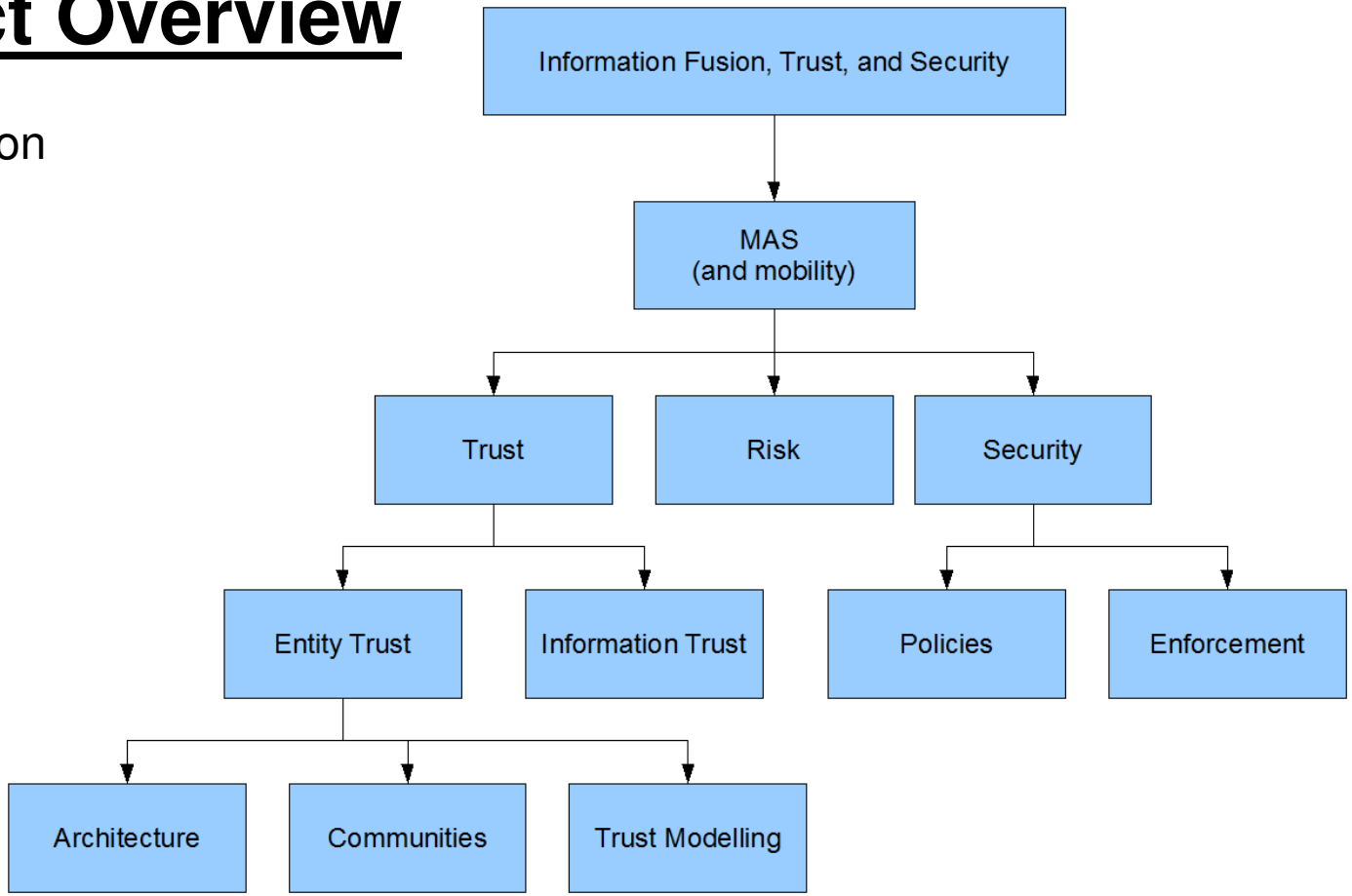
16th October 2008

Itinerary

- Mobile Agent Systems
- Remote Entrusting
- Existing Approaches for Trust
- Importance of Architecture to the Deployment of Remote Entrusting
- Architectural Design
  - Centralised
  - Decentralised
  - Hybrid

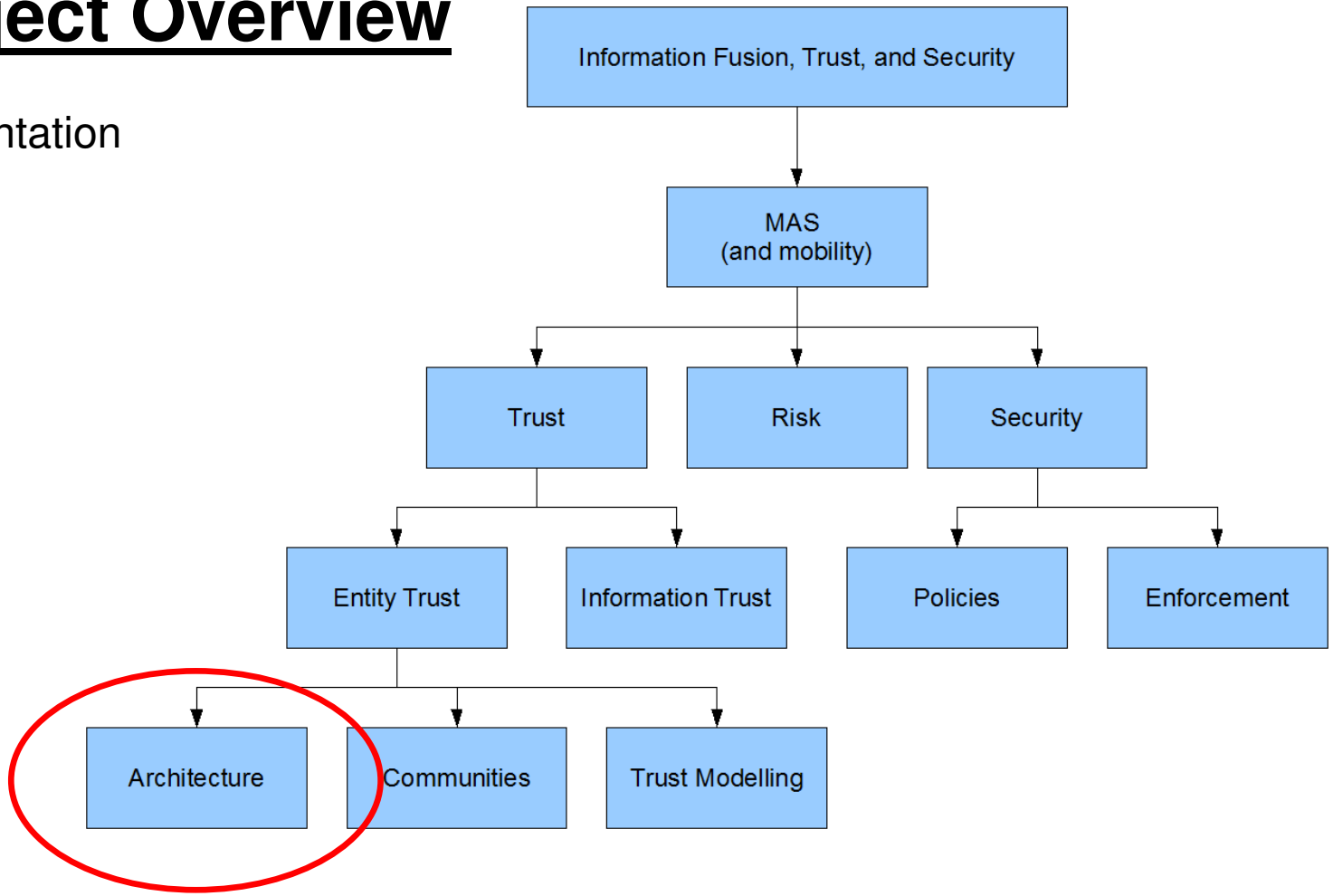
Project Overview

Presentation
Scope



Project Overview

Presentation
Scope



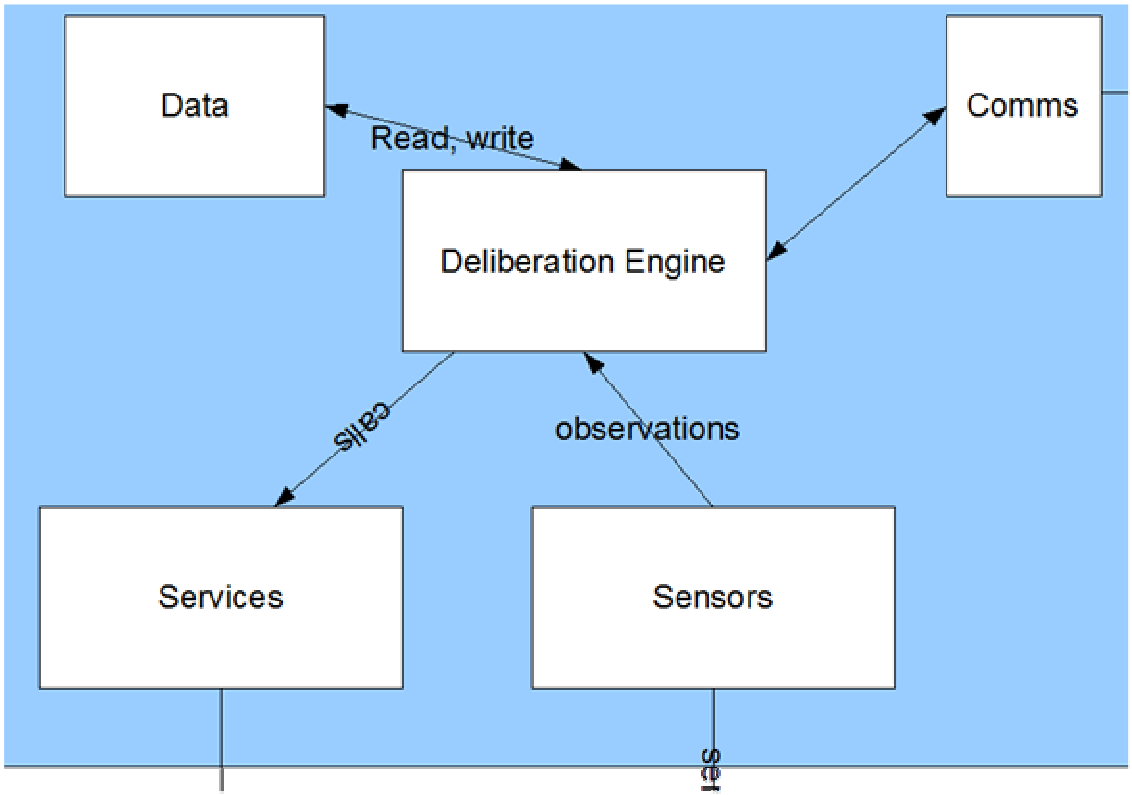
Mobile Agent Systems – Why?

- **Reducing Network Load -** Reduce network interactions
- **E –Commerce –** Purchasing Agents can easily be dispatched to find, barter and buy goods.
- **Network Management & Asynchronous Execution -** Mobile devices (such as PDA's) can be turned off after an agent has migrated and the agent is still able to perform tasks within a network. *It is then possible (if an 'agent-garage' is in place) for the agent to return to the device upon reconnection to the network.*
- **Resource Utilisation -** Processing capabilities can be shared or transferred to devices more equipped to the task.
- **Autonomy** / **Intelligence –** Agents can make decisions, communicate with their environment and process information)

Mobile Agent Systems

- Agents consists of code, data, execution state
- Autonomous
- Mobile agents can migrate from one host to another and continue execution within the new environment
- Mobile agents are usually small

Mobile Agent Systems – Agent Architecture



Remote Entrusting

- Enabling software components to operate in untrusted environments
- Software based approach as opposed to the hardware approach proposed by TCG and Microsoft NGSCB
- Core of trust entities within network
- Proactive trust (avoiding malicious behaviour)
- Need for trust due to relinquishing control

Remote Entrusting – Using Trust

- Types of Trust:
  - Direct (Observed)
  - Indirect (Recommendation)
  - Reputation
    - Collaboration (Community Level Trust)
- Elements of Trust:
  - Trust is always in relation to a specific action
  - Trust is based around one or more observable properties

Existing Approaches

- These generally fall into two categories
  - Access Control Mechanisms
    - Trusted hardware
    - Code Obfuscation and Encrypted Functions
    - State Appraisal
    - Black / White Box
  - Trust Modelling
    - TRAVOS
    - TRUMMAR
    - QoS Selection with Trust and Reputation (Vu et. al.)
    - ….

Architectural Considerations

- Assumptions:
  - The presence of trusted hardware is not guaranteed thus, we assume no trusted hardware exists
  - Mobile Agents within the architecture remain small, highly mobile entities
  - Decision making is undertaken by autonomous entities
  - Malicious behaviour exists and is observable within the system

Architectural Considerations

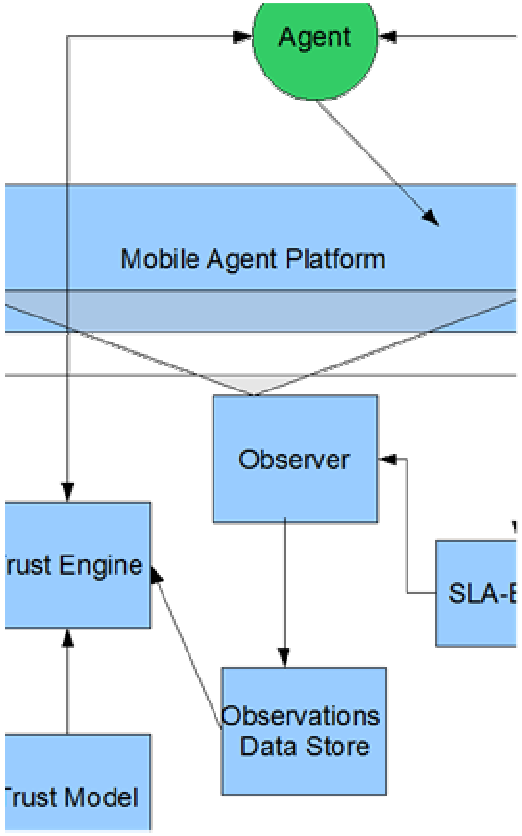
- Requirements:
  - Agents and Hosts (Entities)
  - Observers
  - Measurement for Expectations
  - Trust Deliberation / Management Mechanisms
  - Conflict Management Mechanisms
  - Data Management

Architectural Design

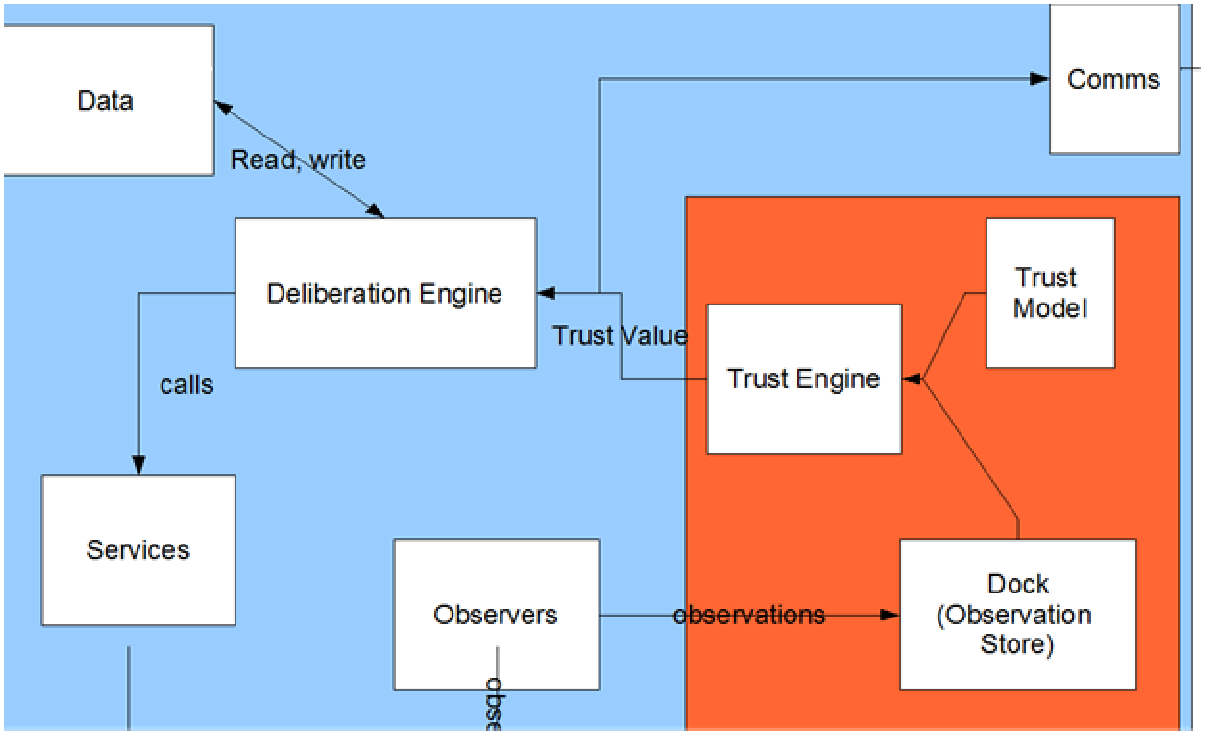
- Centralised
  - Platform provides all services required to utilise trust
- Decentralised
  - Trust mechanisms encapsulated within agents and distributed
- Hybrid
  - Trust mechanisms embedded within the platform and distributed within agents

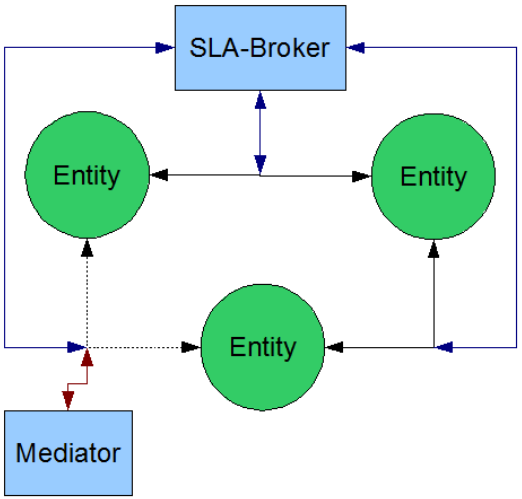
Centralised Architecture

- Agent platform observes the behaviour / interactions of entities (omnipotent observer)
- Observations are stored in a centralised data-store (at least conceptually)
- SLA produced by broker to establish expectations
- Centralised trust engine generated reputation information – shared by all entities



Decentralised Architecture

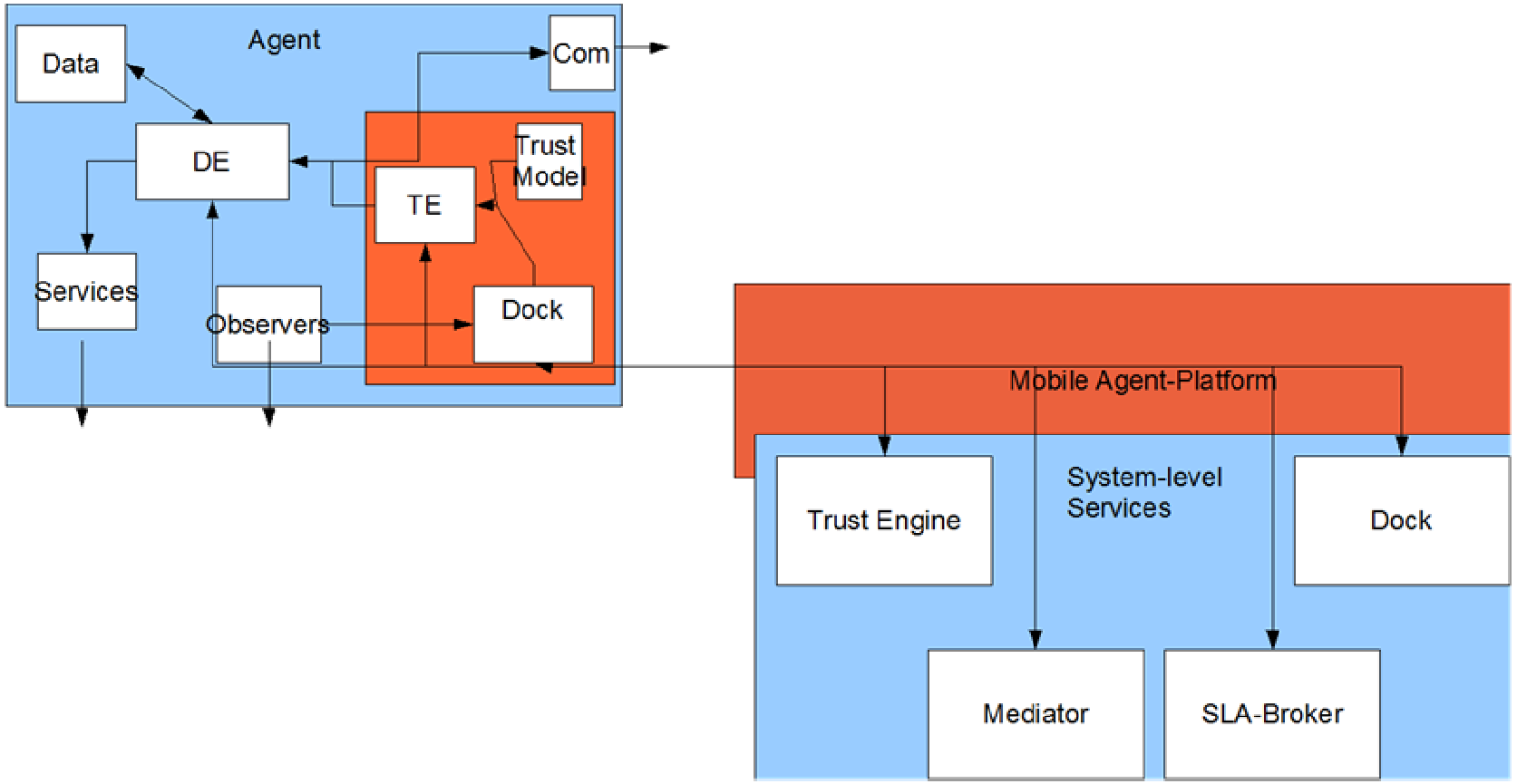




Decentralised Architecture

- Encapsulated within the agent:
  - Trust Engine – Deliberates over known observations
  - Trust Model – Subjective and equips individual agents with the ability to manage trust and apply a 'Trust Value'
  - Dock – Provides mechanisms by which each agent can access their trust information, this is stored remotely and synchronised (in full or in part).
  - Mediation is undertaken by a third-party (agent) agreed to be trustworthy in such a task by both entities of the original interaction
  - SLA-Brokers exist as agents (another service) for entities to use in establishing expectations.

Hybrid Architecture



Hybrid Architecture

- Trust deliberation remains within individual agent, thus, enabling individual, and recommended trust
- Platform can provide services for reputation trust
- Dock synchronisation can be achieved using a dedicated services or a centralised dock
- Mediation and SLA-Brokering is provided as services by the platform for use by individual entities

Architecture Comparison

	Centralised	Decentralised	Hybrid
Observations	Reputation	Direct, and Indirect	Direct, Indirect, and Reputation
Subjectivity	None	High	High
Trust Properties and Trust Model	Predetermined	Subjective	Subjective
Scalability	Questionable	High	?

Future Work

- Elaborate on initial empirical results w.r.t Architectures
- Scalability and usability testing
- Incorporate ‘risk’ factors into the deliberation
- Compose trust, policy and security elements
- Formal underpinning and link between architecture, model, and trust.

Conclusions

- Presented three architectures for use with trust implementation
- Maintain requirements for mobile agents expanding this with trust
- Careful consideration must be given to the architecture when utilising trust, in addition to the modelling of trust itself