Defence and Communications Systems

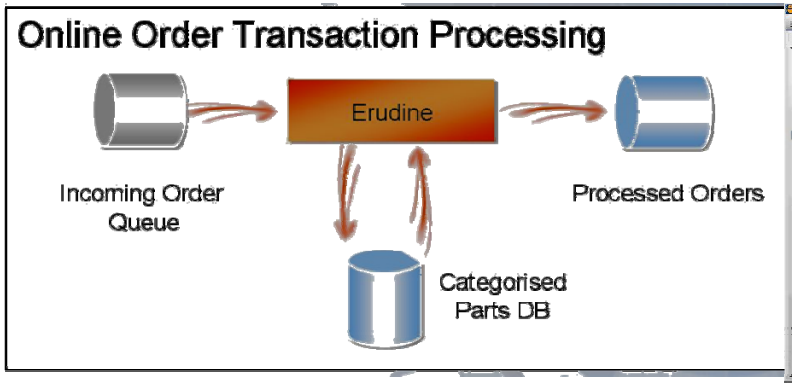
The EADS Systems House

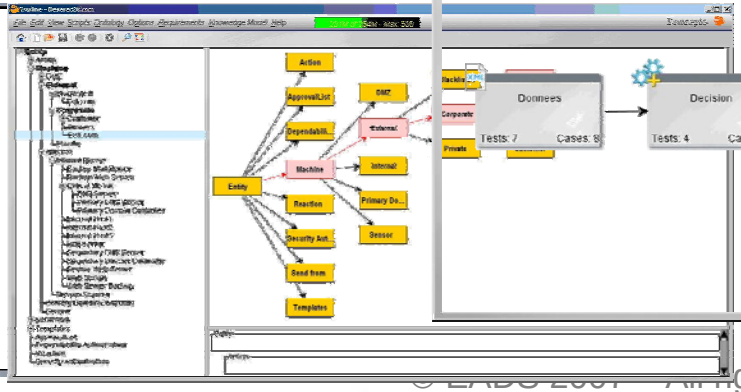
EADS and Erudine Behaviour Engine

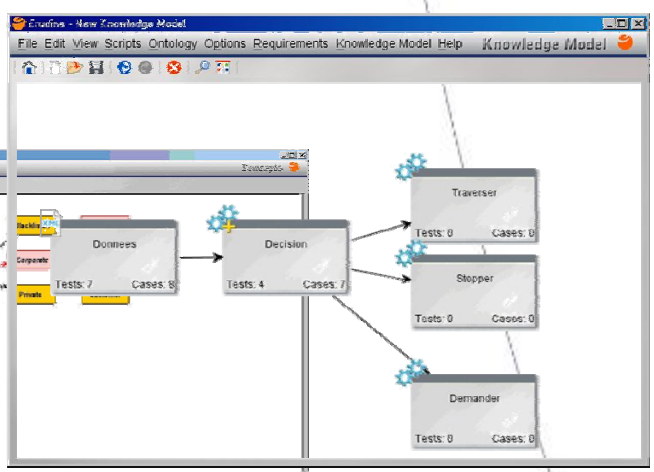
A standard way to manage the business rules within EADS Defence market

What is Erudine Behaviour Engine?

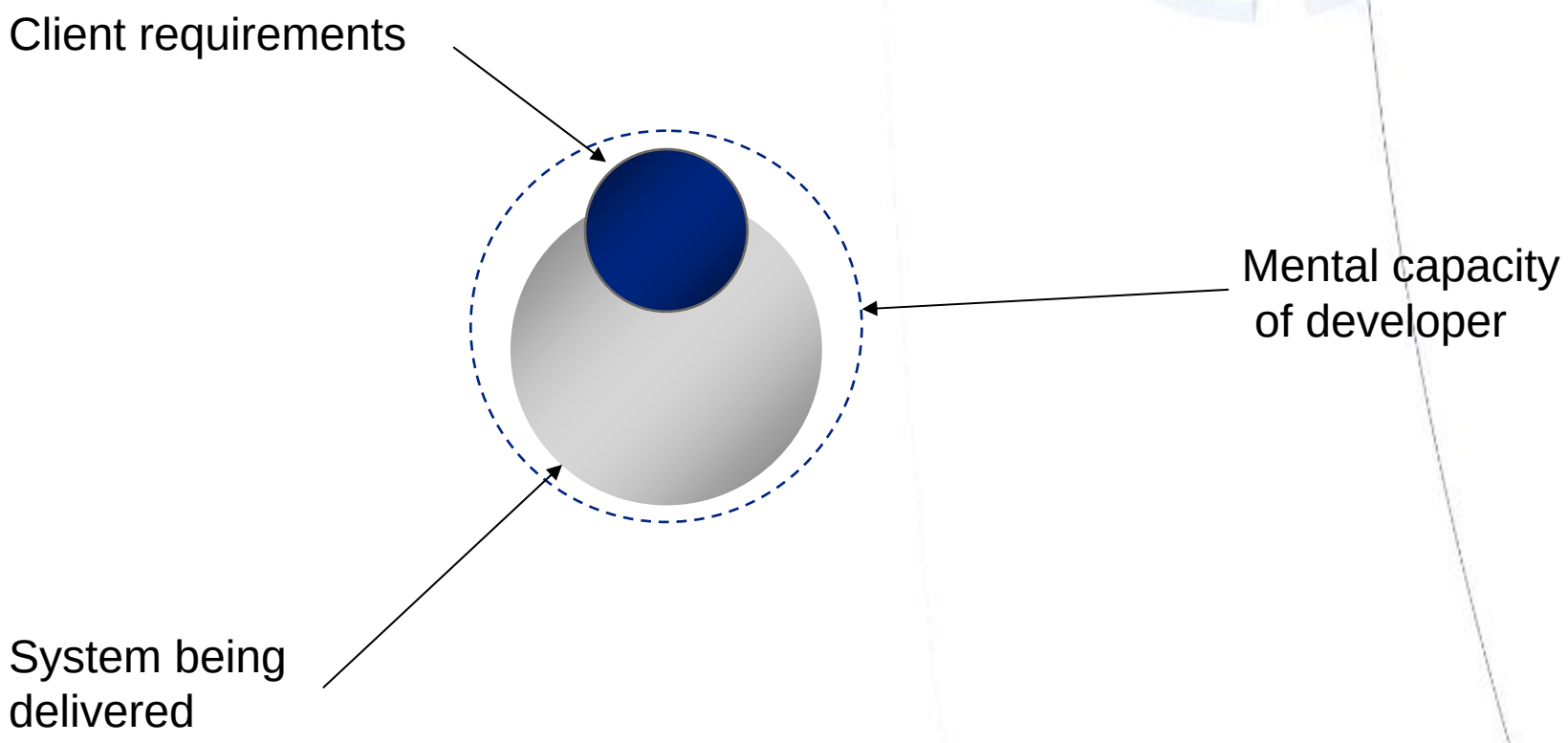
- Erudine Behaviour Engine is a revolutionary way of
  - **capturing the behaviour of systems** to reduce large elements of software development
  - **automate the work** carried out by IT developers
- Erudine Behaviour Engine addresses **complex** systems development that represent areas of problems and **risk**
- EADS is now embedding Erudine technology within projects – teaching the business rules



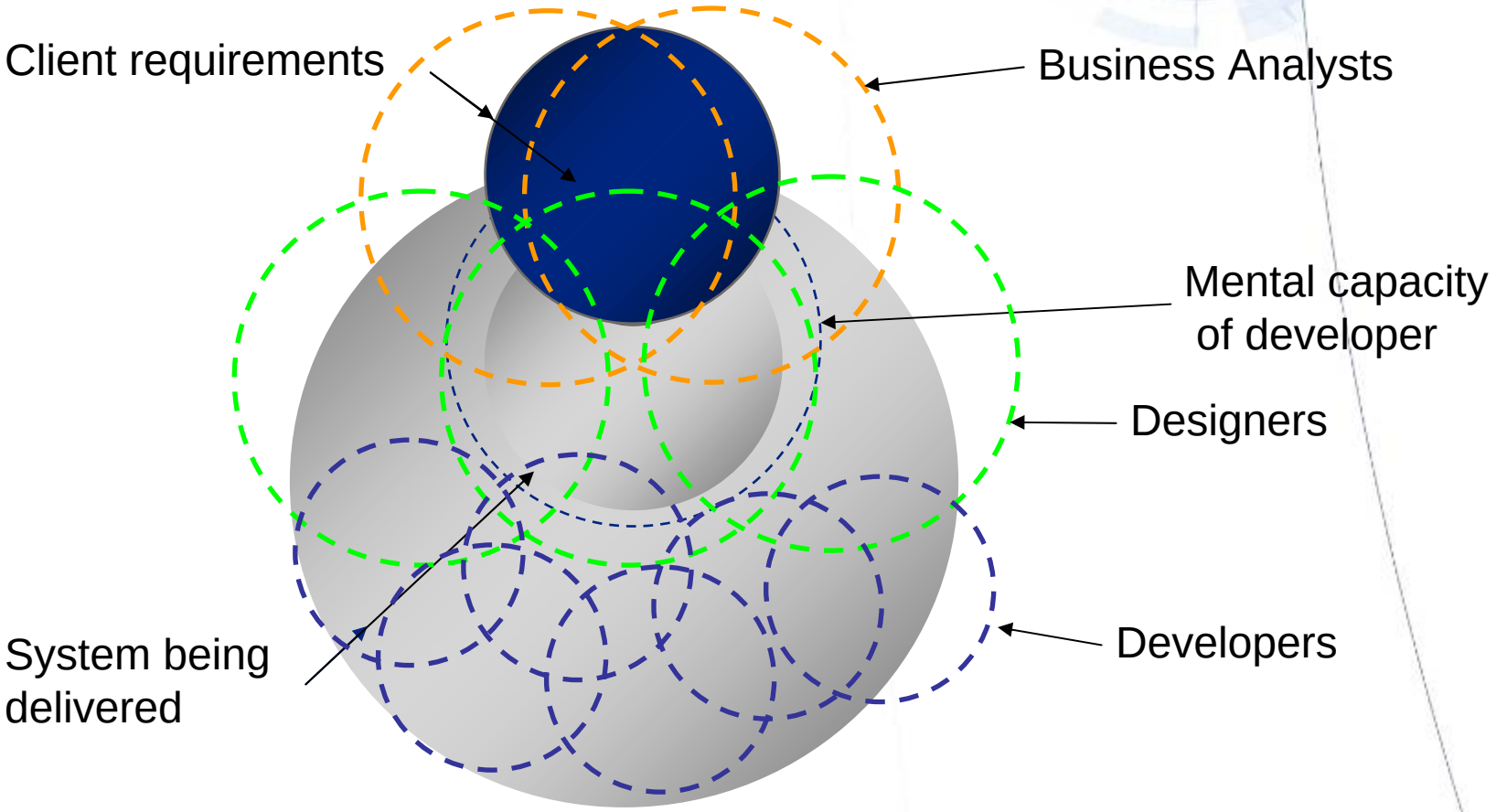




Why are legacy systems created ?

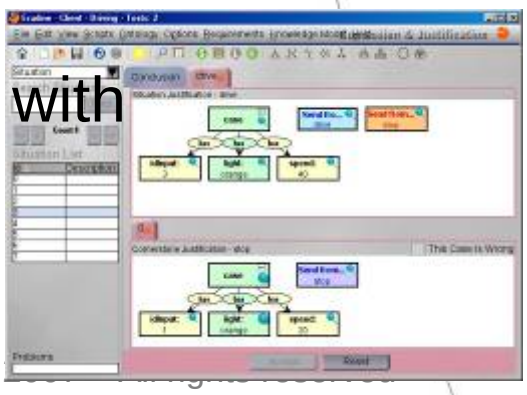


Larger & more complex projects require more people equals more communication

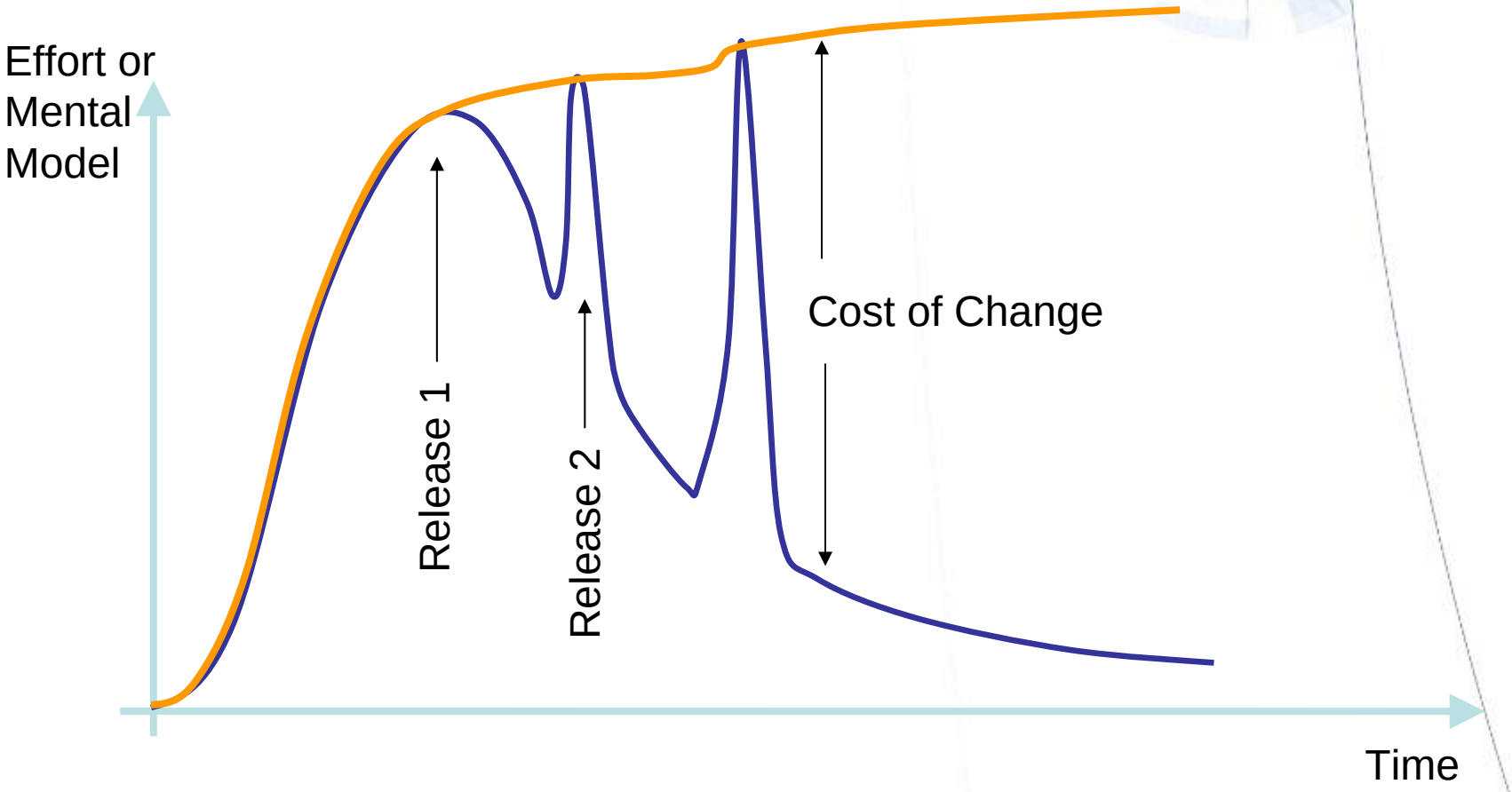


Key technology aspects of Erudine Behaviour Engine

- Innovative rules management, ordering and processing
- Data and rules represented as **graphs**
  - Enabling new methods of processing data
- Captures **Tacit knowledge** (i.e. use cases that are made from many rules) and uses corner cases to interact with the expert.
- **Intuitive graphical user interface** and graphical data representation operating in near real-time;
  - Enabling new ways of interacting with a development system.
- Automatic **impact analysis** and interaction with corner cases



What are the results of being able to “teach” ?



How to use EBE for managing business rules?

- To capture the business rules
  - Based on examples (case based reasoning)
- To ensure consistency
  - Automate the work
  - Ability to test under simulations
- To check the compliance to the business policies
  - Or to any requirement document
  - Or within groups of experts

- Applications are developed
  - by embedding the automation of business rules
  - operational people can teach the system what to do

- Update of the business rules
  - Integration of operational feedback
  - As soon as the regulation changes

Interoperability

- Business interoperability
  - Tacit knowledge is captured and reusable
  - Iterative capture – update between projects
  - Outputs are
    - Implementation
    - Requirement documents
    - Corner cases
- Technical interoperability with other systems is ensured both :
  - in design phase for standard requirements management (Doors,...)
  - in implementation phase by mean of standards for development (databases, java, web services, messaging...)
  - in acceptance tests (Doors, test suite)

Example of application: Serket
SEcuRity KEeps Threats away

- Security of places and public events
  - data coming from discrete devices
  - automatically correlated and analysed
- Demonstrate the new architectural principle
  - Open security platform for public places and events
  - Complex Event Processing
- EADS Threat Assessment Module
  - recognize situations
  - to correspond to threatening behaviours
  - view and control the decision process
  - http://www.itea2.org/

Thank you for your attention

Questions?