End-to-End Security on the Internet

Esprit Project #20563

http://www.e2s.com

Delivering Secure Electronic Transactions for Business-to-Business Electronic Commerce over the Internet

Andrew Herbert
Project Technical Director
APM Ltd

http://www.ansa.co.uk ***ajh@ansa.co.uk***

8 September 1997

With thanks for my
E2S colleagues for their slides!

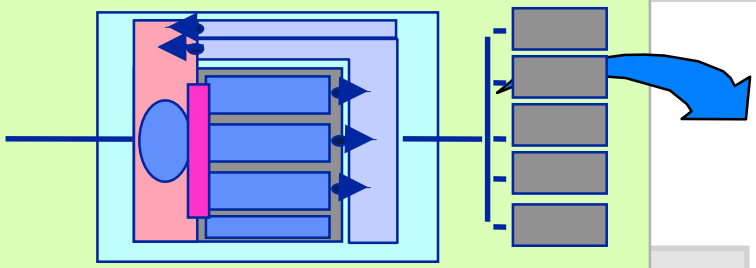
E2S - End-toEnd Security on the Internet

http://www.e2s.com

35 Person-Years over 2.25 Years...

Developers & exploiters GemPlus, HP Labs, Onyx Internet, GMD, HP-ENSD, APM, VISA

Technology Development



User-Lead Trials

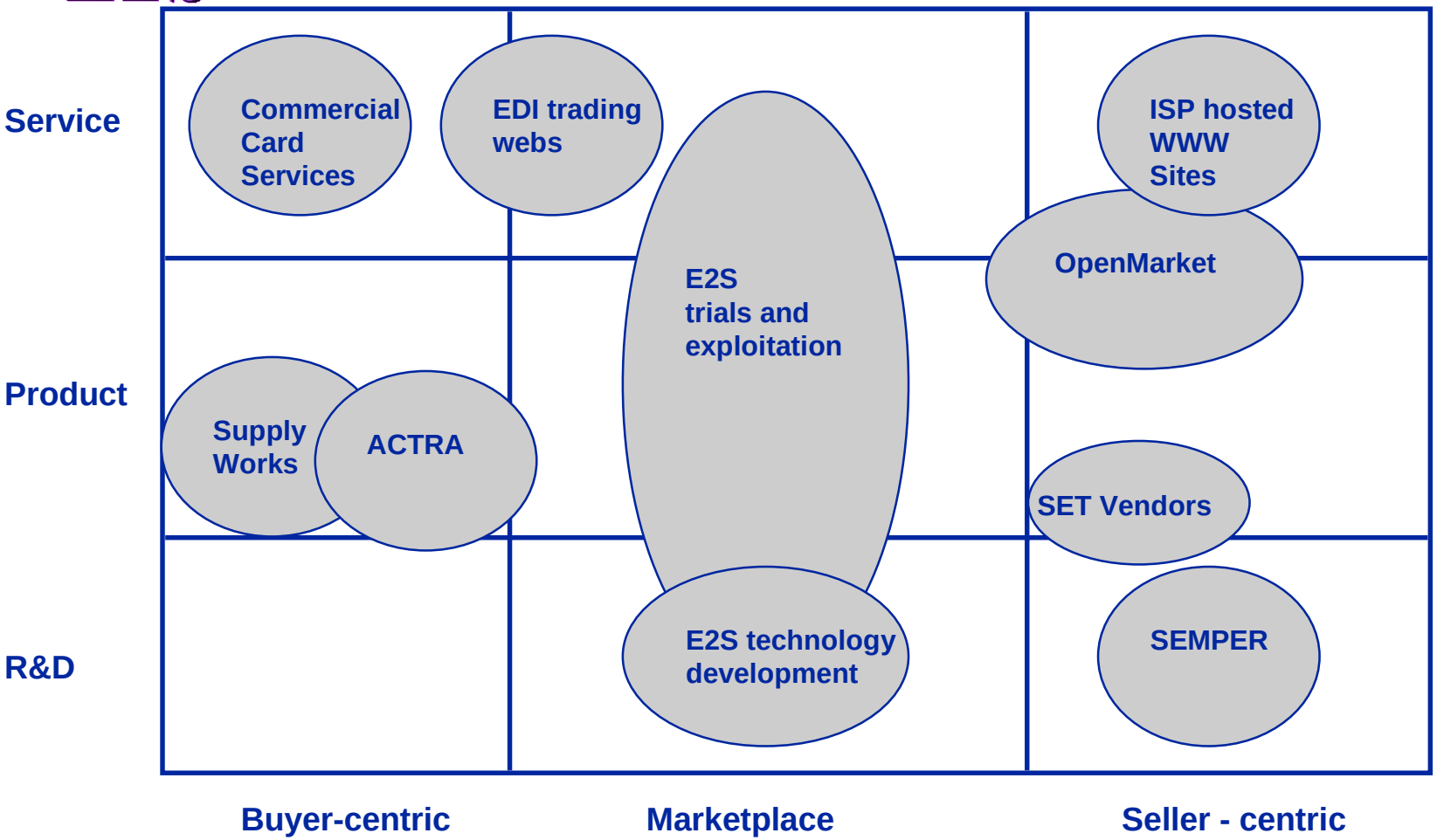
User Group

Swiss Bank Corporation - Warburg
HP-WCSO
TU-Berlin
SmartCard Forum

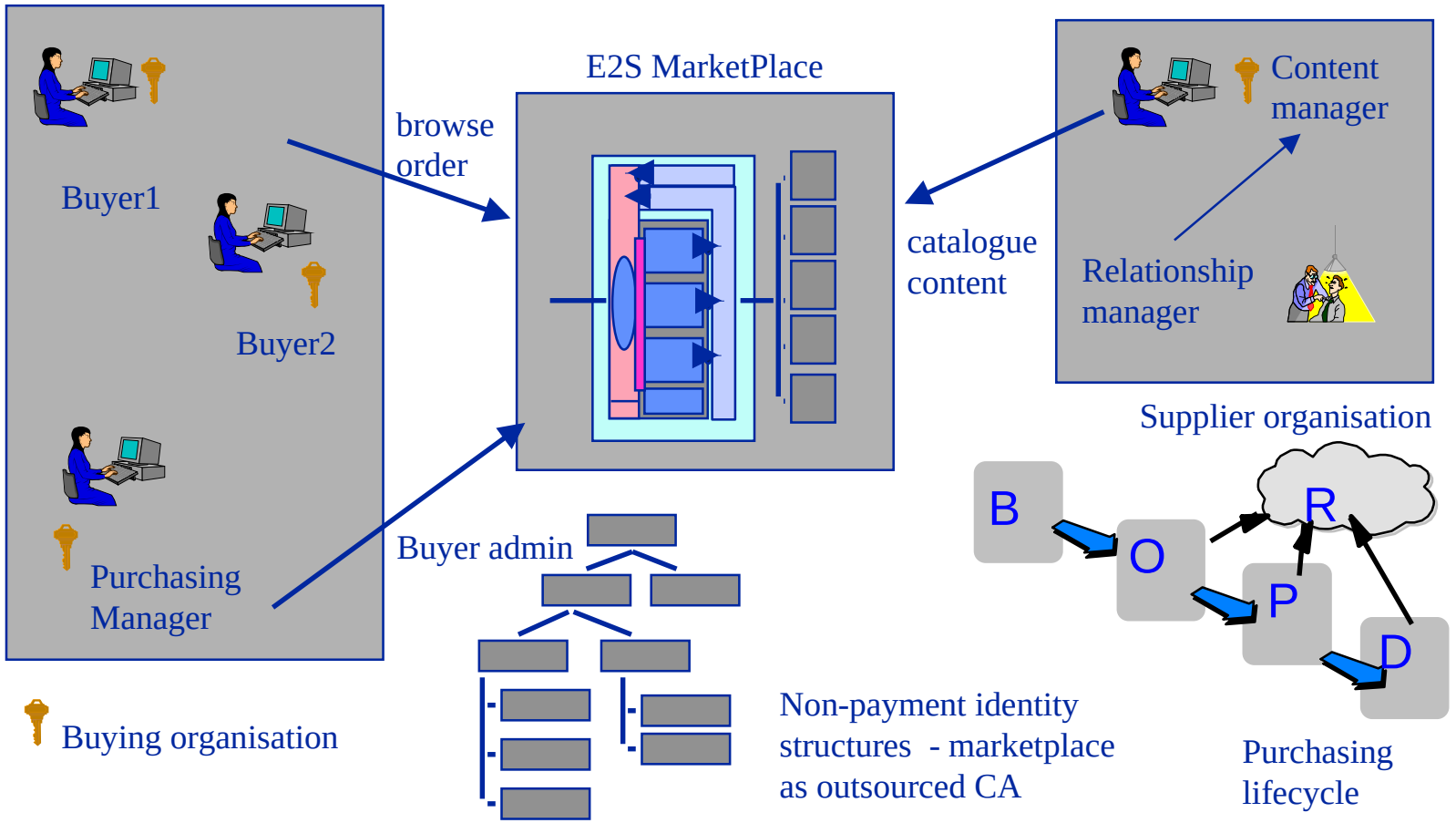
Deliverables

- **Implementation Architecture**
- **Key Components**
- **Proven Secure Infrastructure for:**
  - **Information access and delivery**
  - **Transactions**

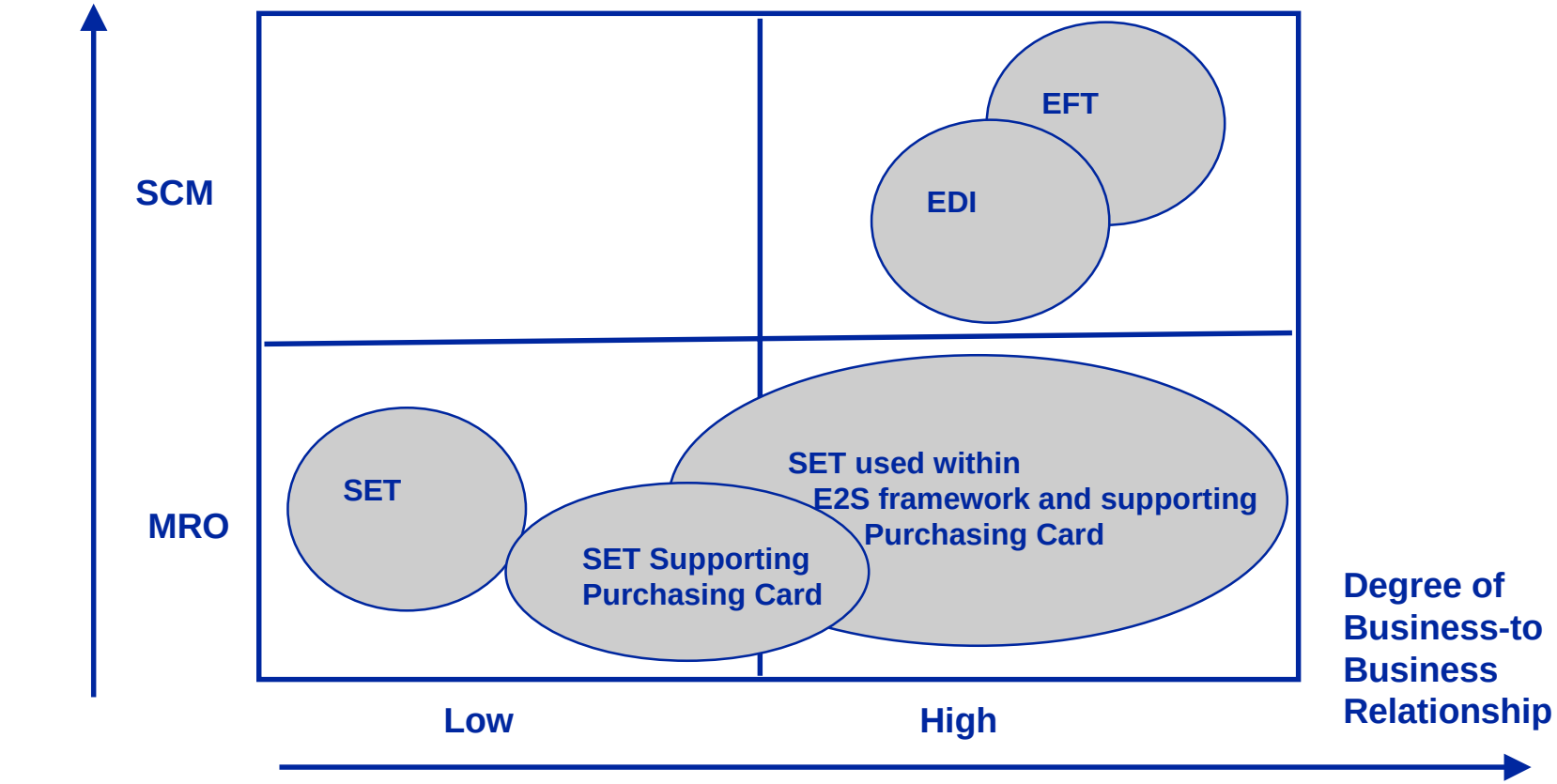
E-Commerce Segmentation



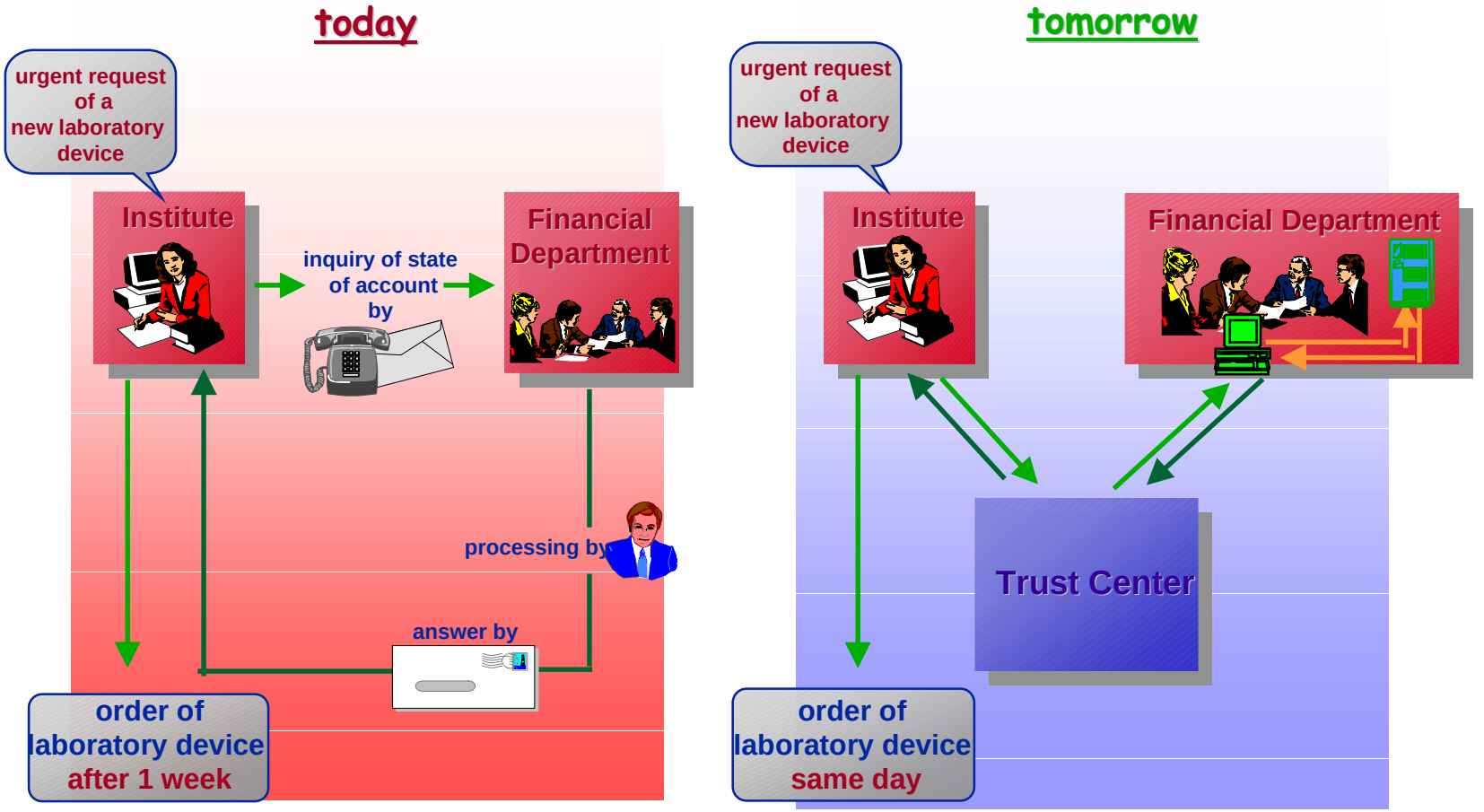
Much more than payment…



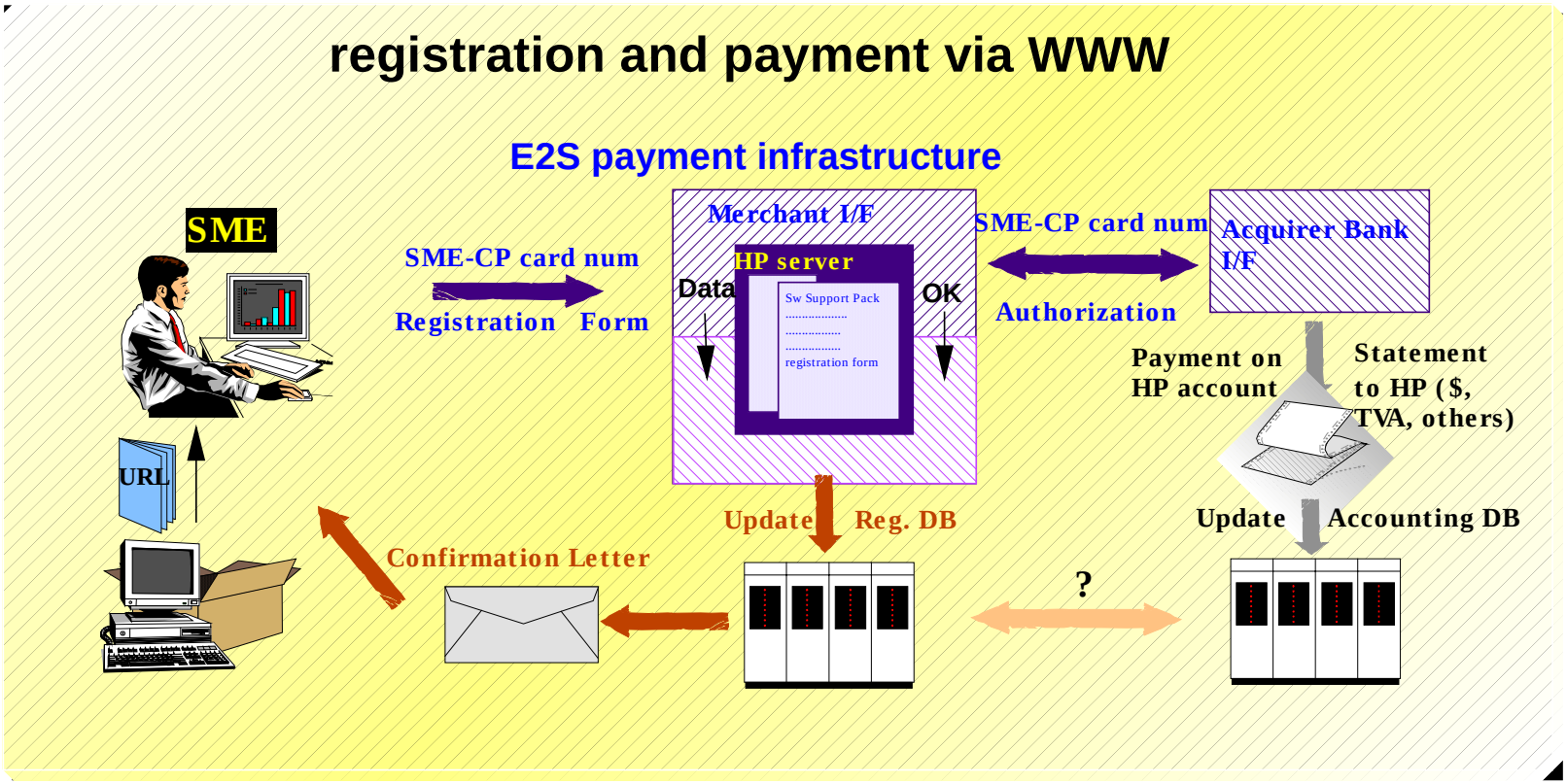
Payment Infrastructures



TU Berlin Demonstrator: Efficient Administration



Hewlett-Packard Demonstrator: Secure Purchase of Software

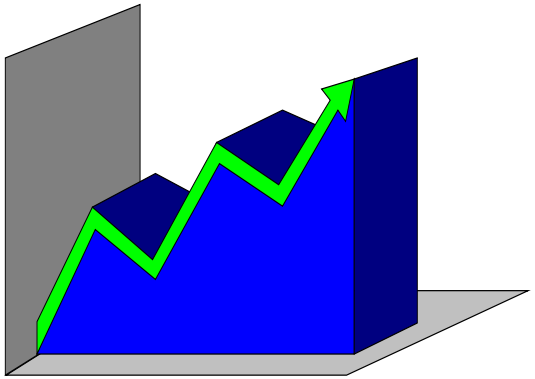


SBCW: Internet Investment Broking

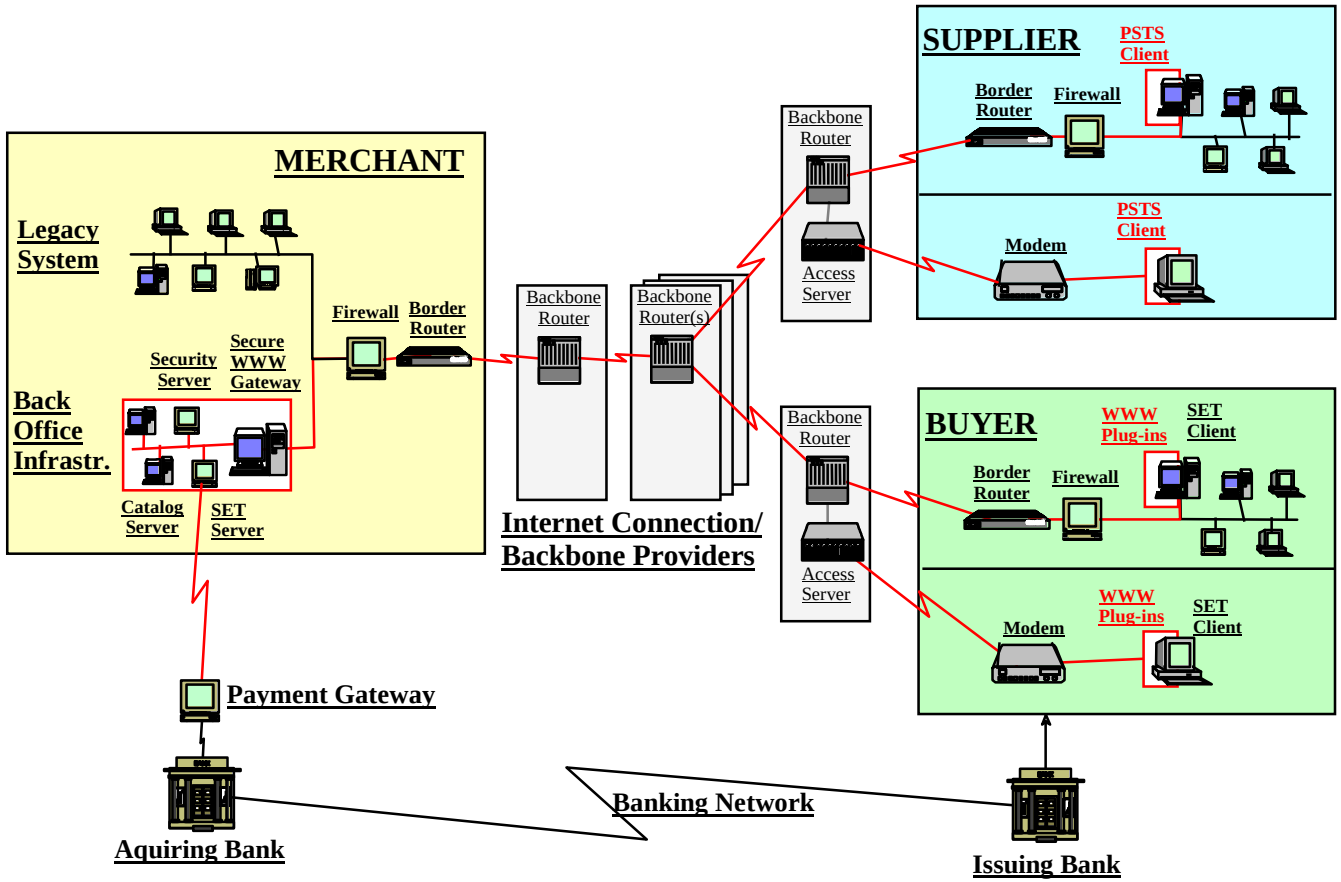
- **Deploy apps atcustomer site**
  - *Drag and drop user interface*
  - *User profile defines services*
  - *Transactions fed to trading floor*

- **Private key management**

- **Reduce costs using internet**
  - *No private network to run*
  - *Faster deployment*
  - *Larger scale deployment*

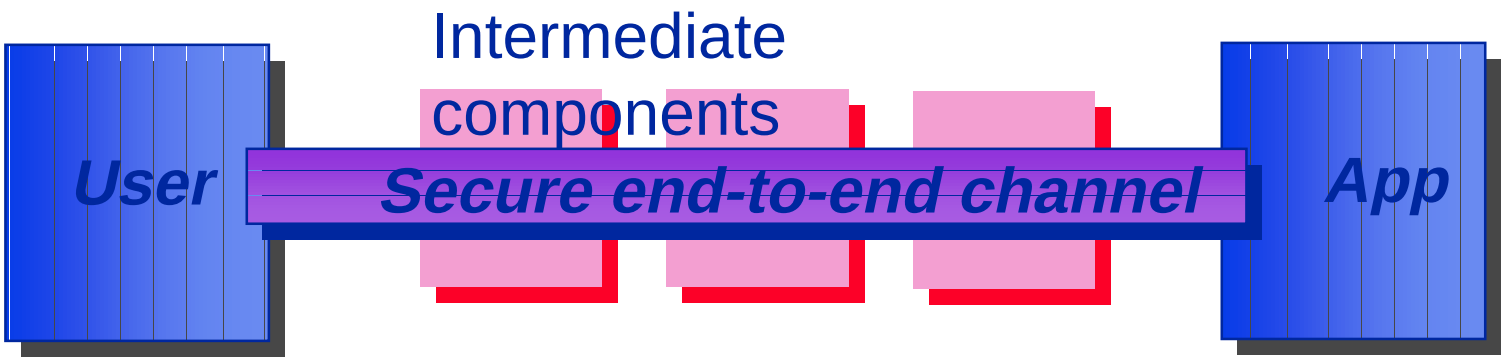


Onyx Demonstrator: Secure Marketplace

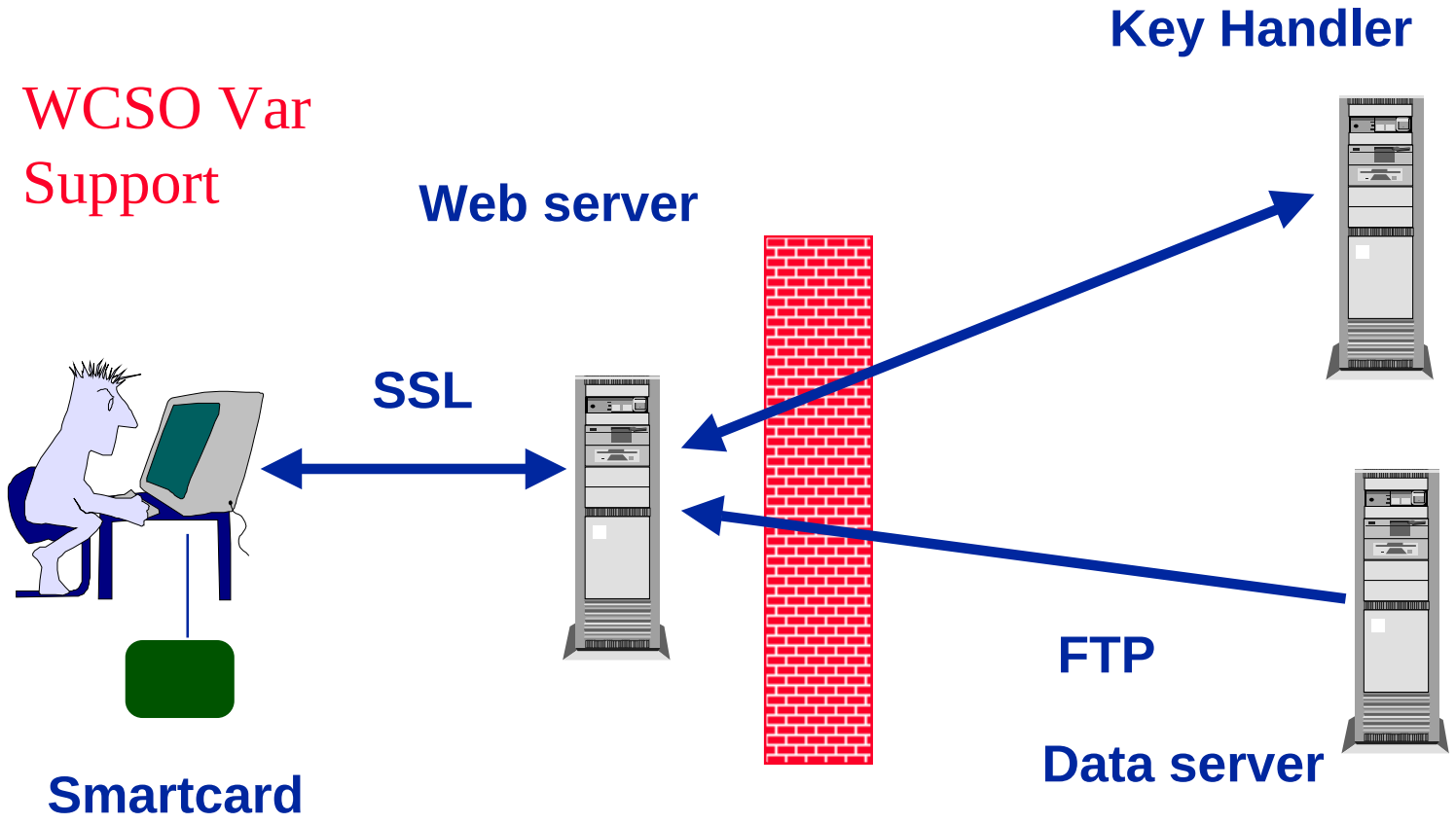


Security From End to End

- Components need to be used to construct secure channels end-to-end
  - Smart cards and system partitioning for physical security
  - Public key infrastructure for digital identity



Model I

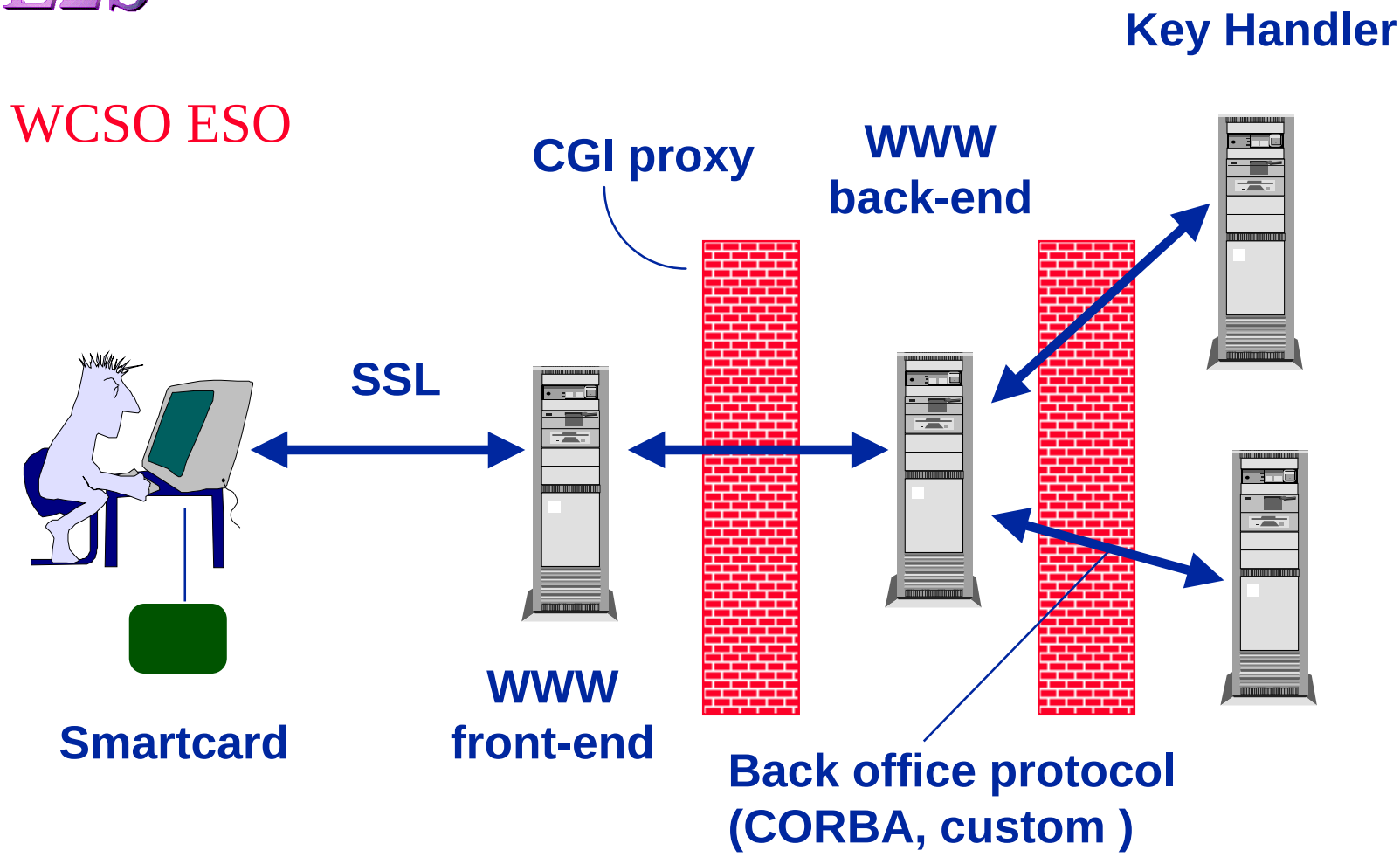


TUB e-mail and PEM equivalent

Model I - Features

- **SSL gives privacy and access control**
- **O.K. for publishing**
  - *prices, stock levels*
  - *“batch mode”*
- **Not transactional**
  - *can’t do queries*
  - *can’t place orders*
- **Web server exposed to attack**
  - *content at risk*
  - *packet filtering helps*

Model II

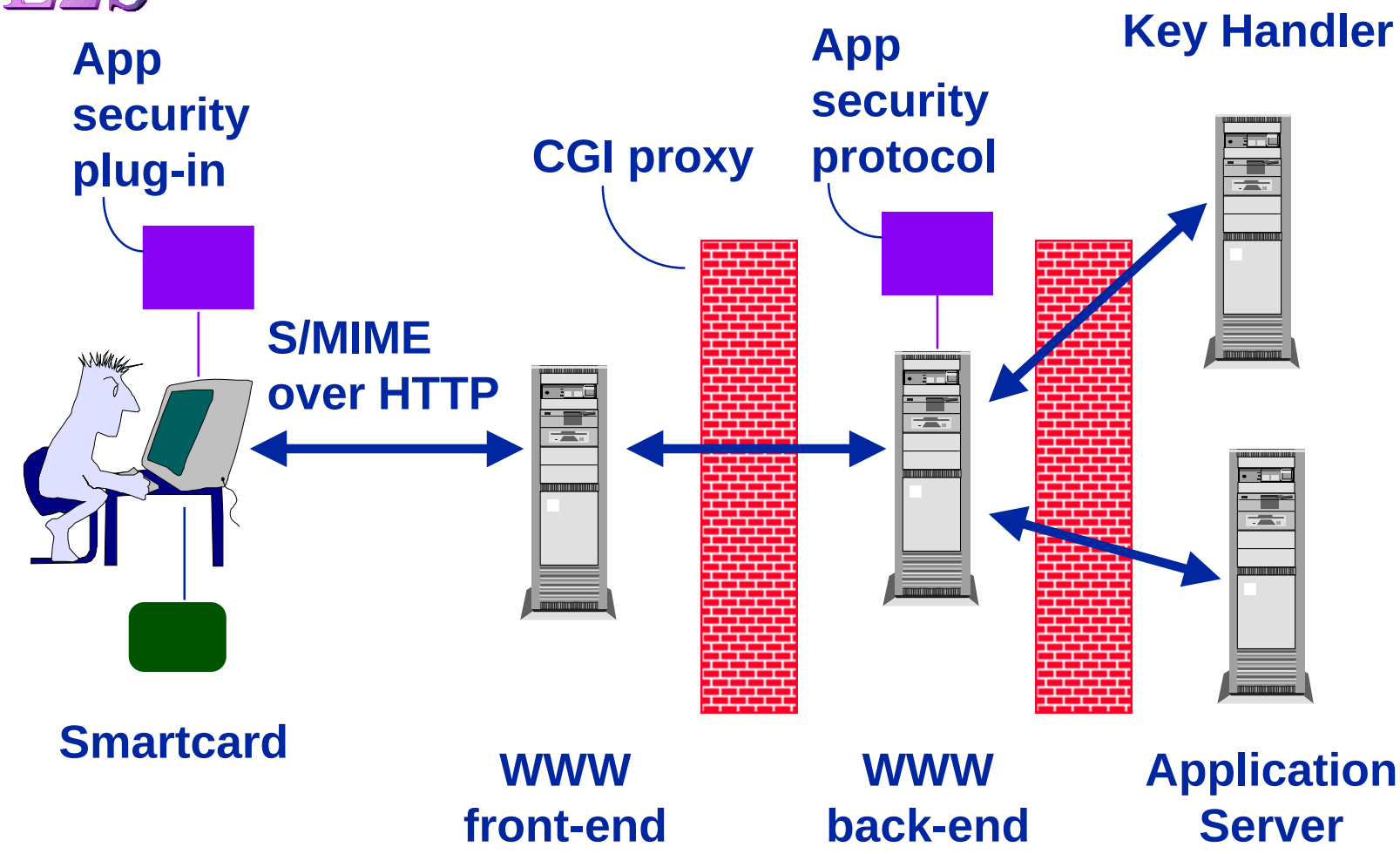

Key Handler
WCSO ESO
CGI proxy
WWW
back-end
SSL
WWW
front-end
Smartcard
Back office protocol
(CORBA, custom)

Model II Features

- **Front-end is simpler**
  - *no content exposure*
- **CGI-proxy at firewall**
  - *filter only allows front-end to back-end connection*
- **Enables transactions**
  - *but only single step*
  - *CGI -> application interface conversion overhead*

Model III

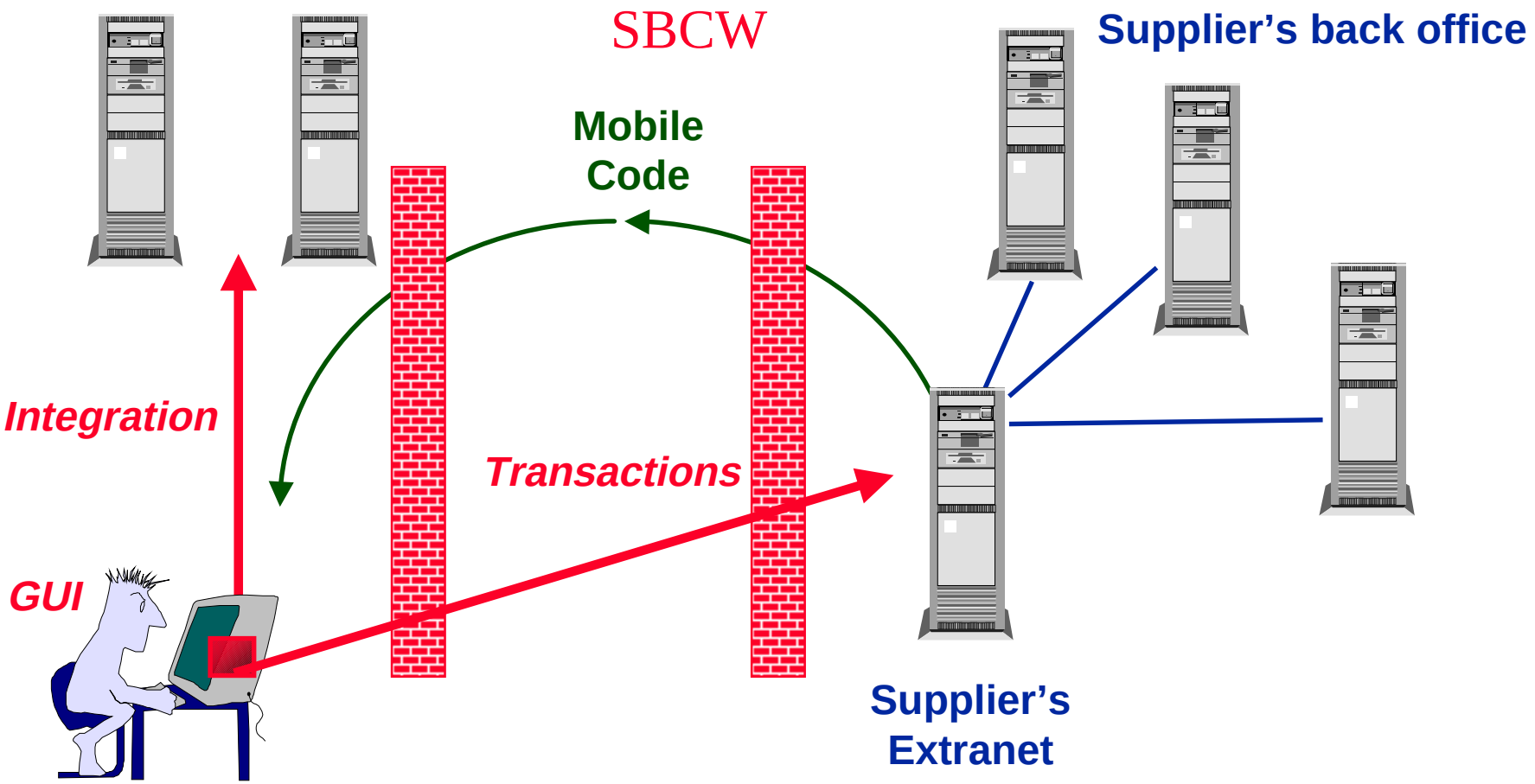
Onyx WSCOLicence



Model III Features

- **End-to-end secure "purchasing protocol" tunnelled through HTTP**
  - ***browse***
  - ***order***
  - ***pay***
  - ***deliver***
  - ***reconcile***
- **But need "purchasing plug-in" at client**
  - *how to distribute this securely?*

Model IV

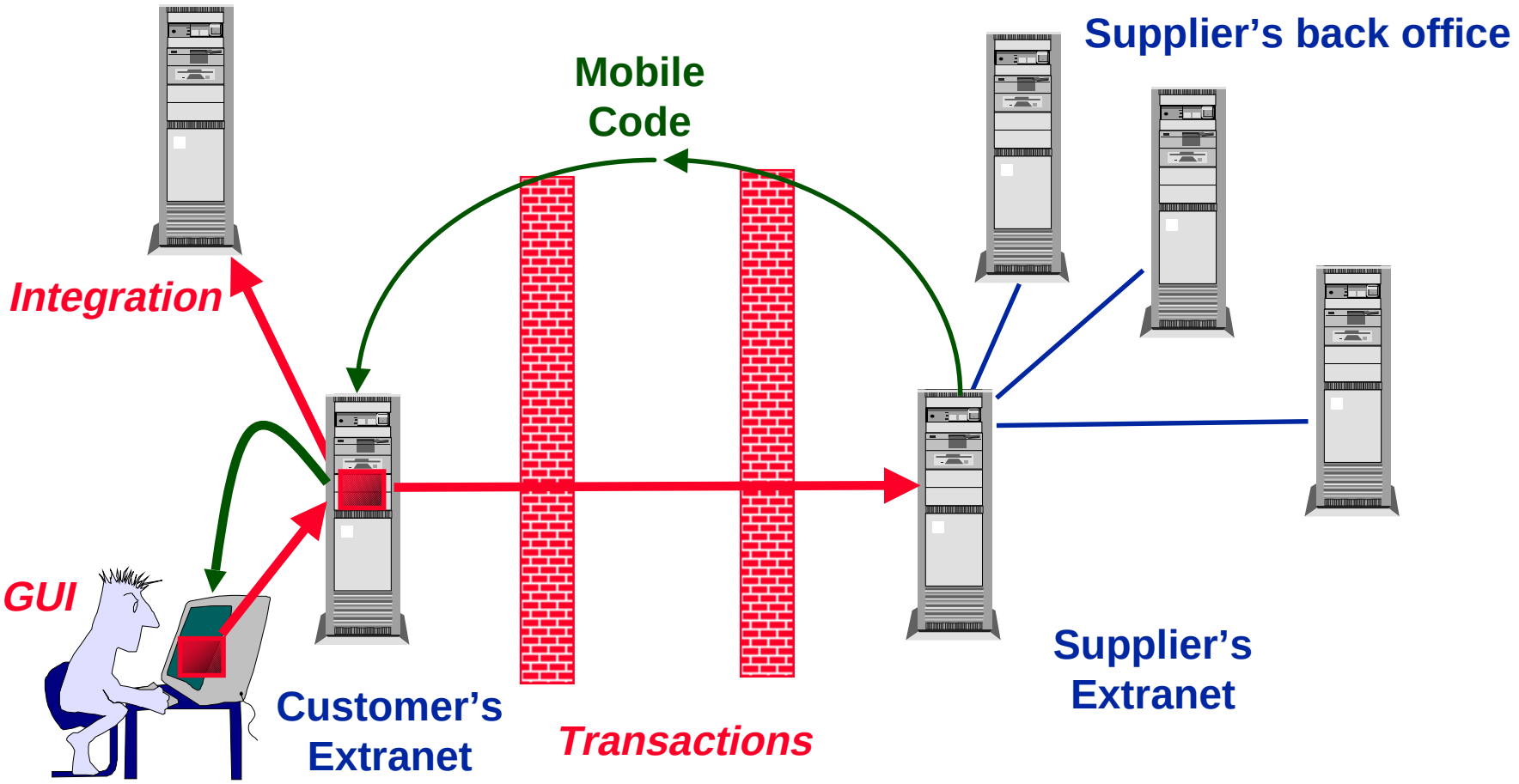

SBCW
Supplier's back office
Mobile
Code
Integration
Transactions
GUI
Supplier's
Extranet

Model IV Issues

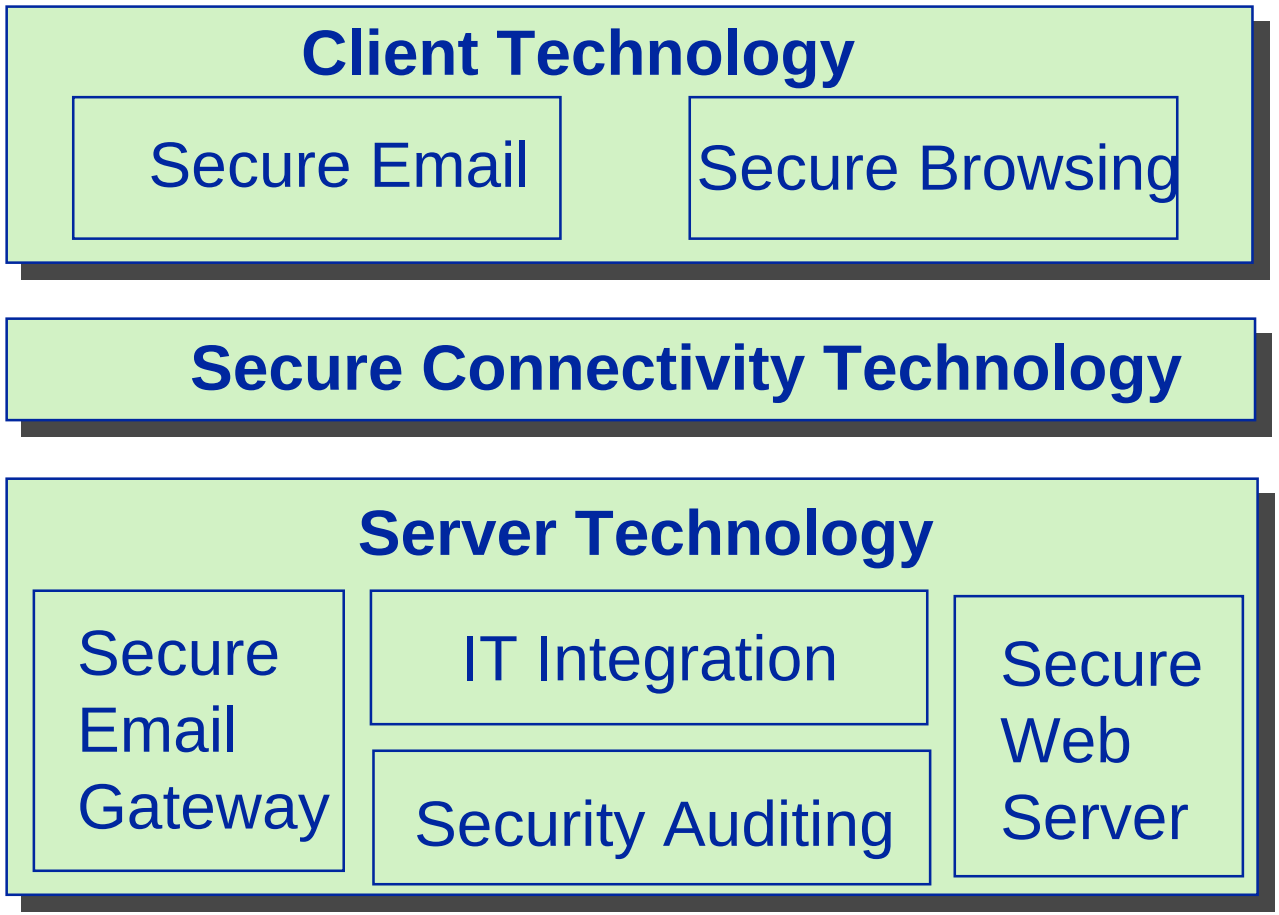
- **Access to Smartcard from Java**
  - **Electronic Wallets hide crypto behind payment API**
- **How does user know it's the supplier's code?**
  - **Digital signature**
- **How to punch back through firewall?**
  - **Java ORBs - IIOP eats ports**
  - **Java Remote Method Invocation (RMI)**
  - **HTTP tunnelling (Visigenic)**
  - **TCP/IP (Iona)**

Model V

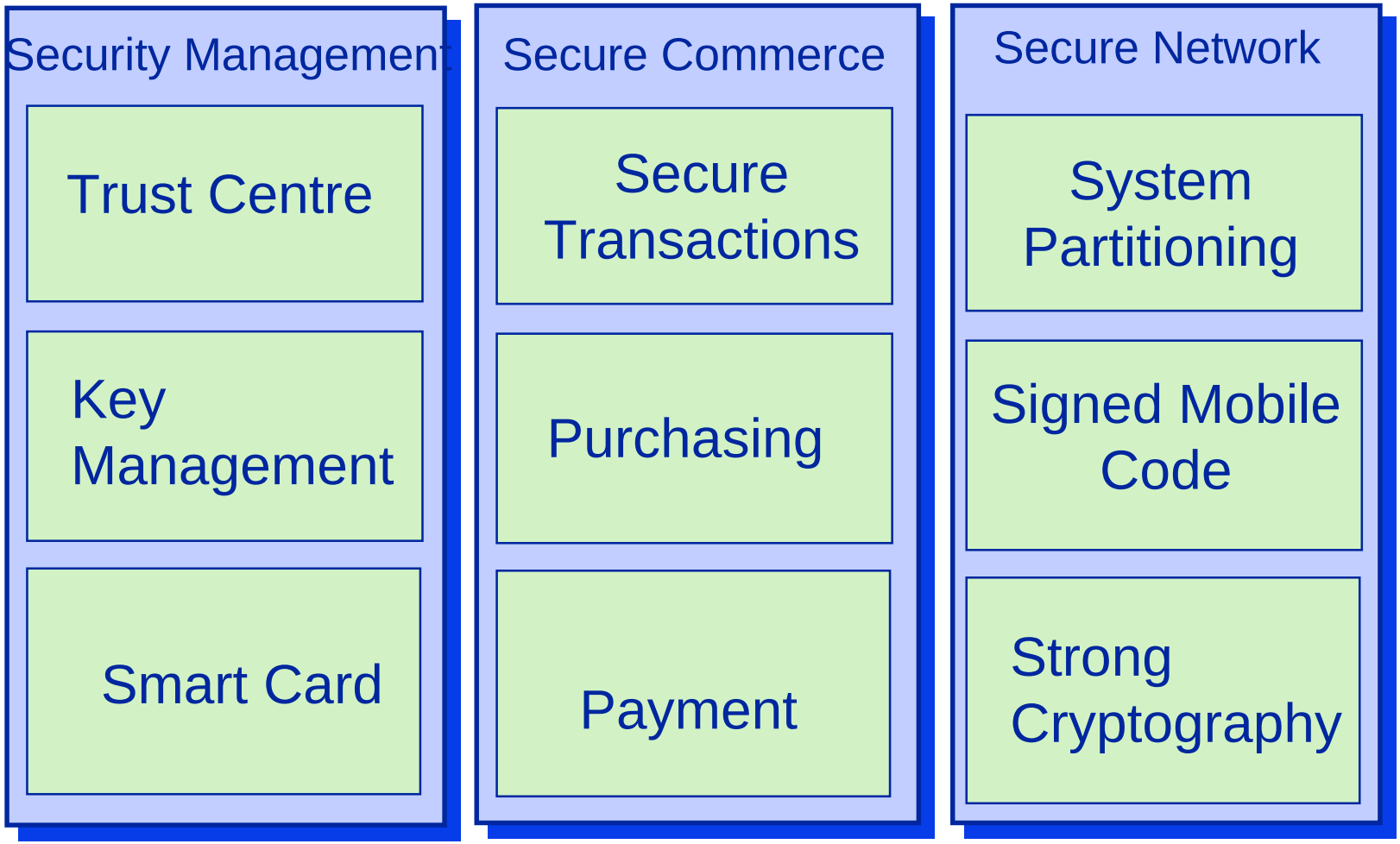
SBCW


Supplier's back office
Mobile Code
Integration
GUI
Customer's Extranet
Transactions
Supplier's Extranet

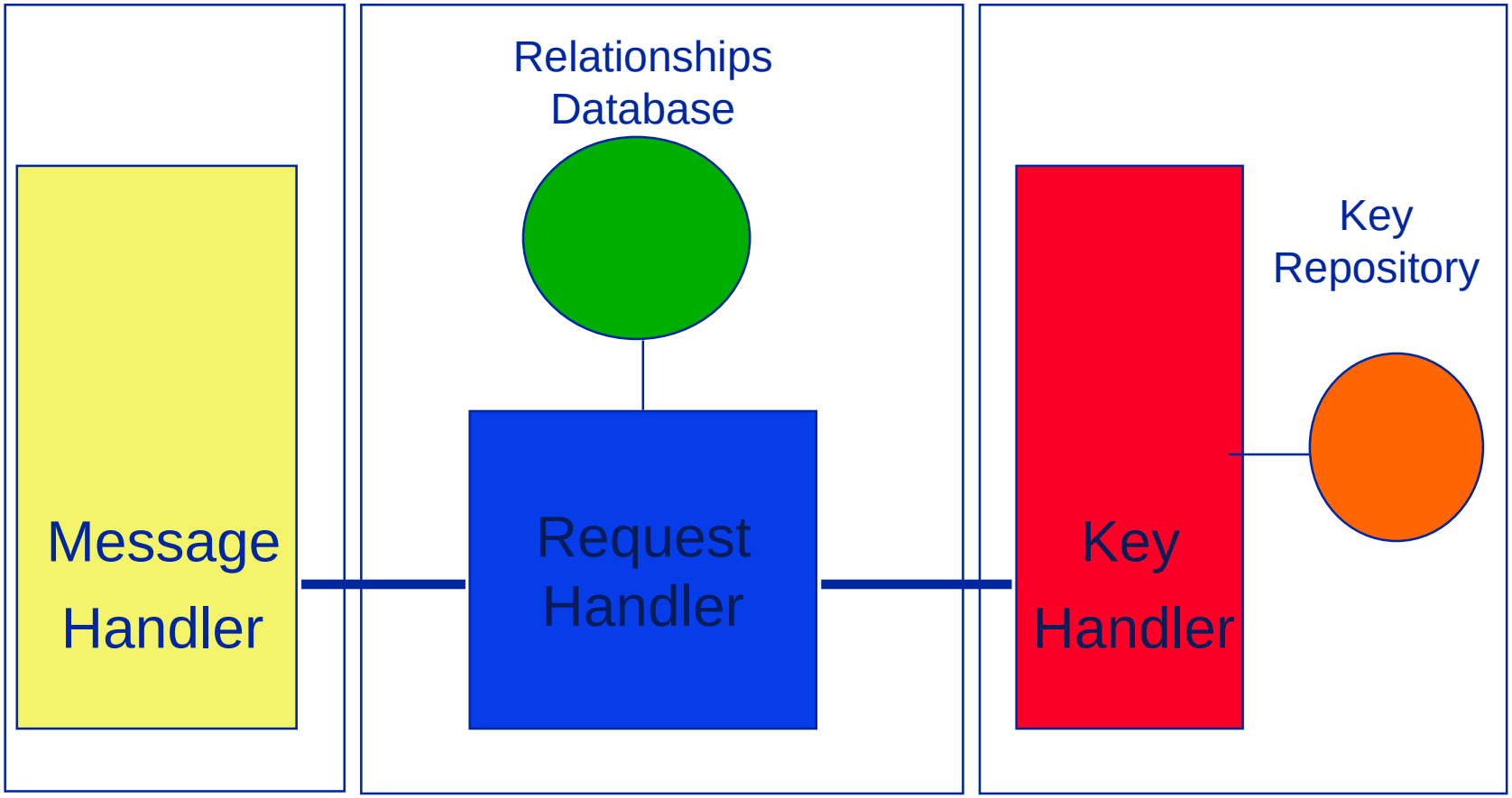
Implementation Architecture



Secure Connectivity Technology



E2S Trust Centre



CMW Firewall

MCGA = multi-compartment gateway agent

policy controlled circuit relay