

Software Escrow Agreements

SS 2013

Stephan Strodl

SBA Research

<http://www.sba-research.org/~sstrodl>

Agenda

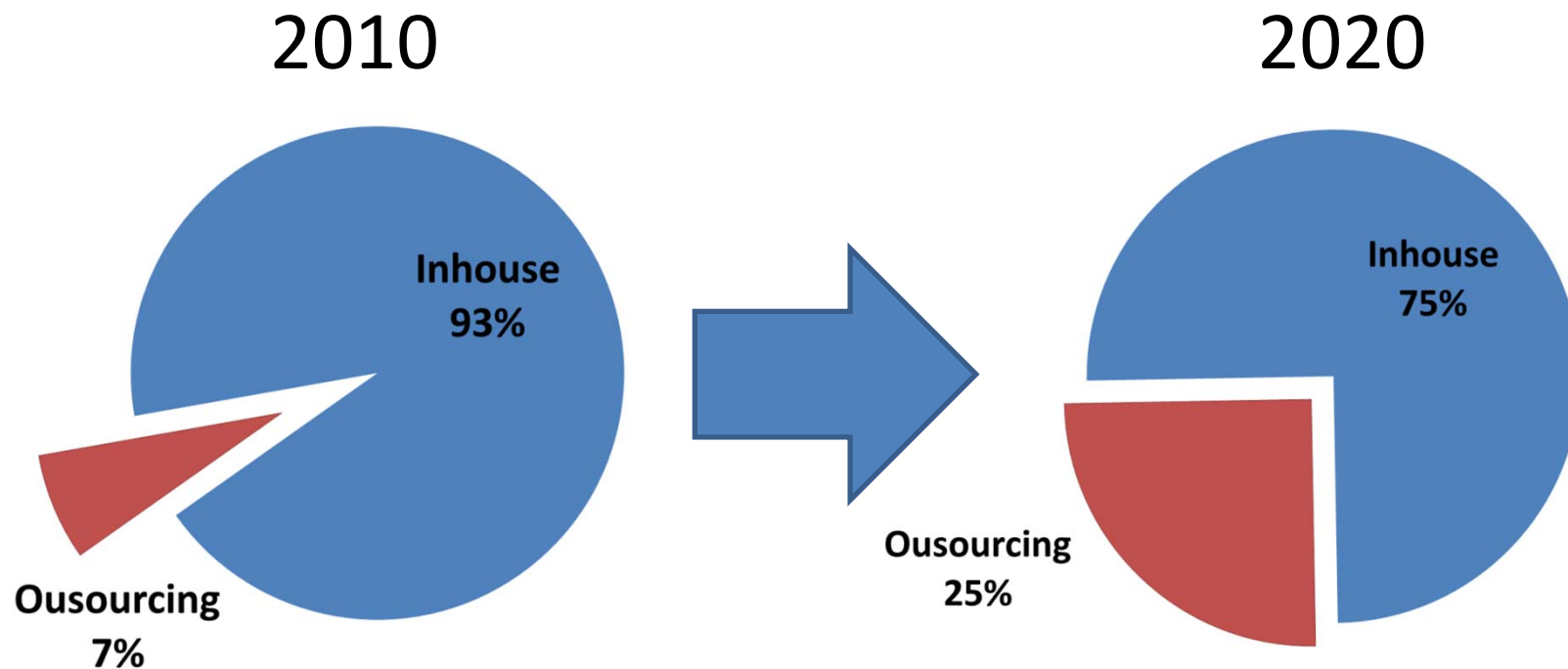
- Motivation
- Risks
- Software Escrow Agreements

Motivation

- Ongoing trend towards IT-Outsourcing
 - Cloud computing
 - Software as a Service
- Outsourcing represents about 40% of the IT-Service Market
- 16% Custom Application Development¹

¹ IDC Austria

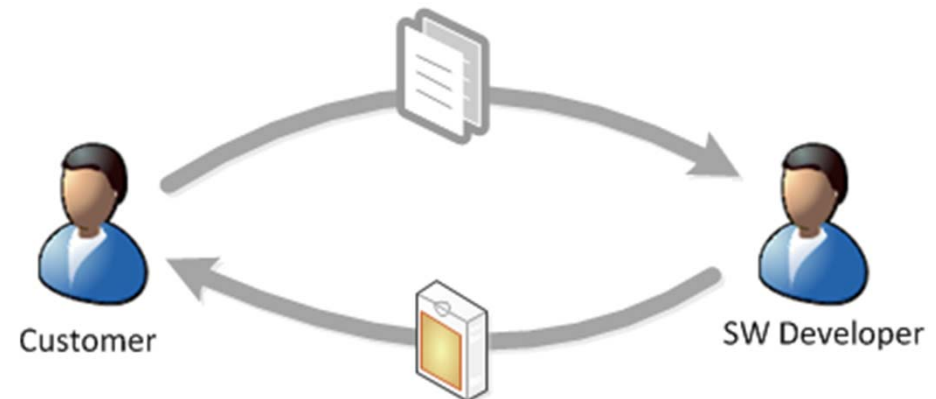
Outsourcing – growing trend



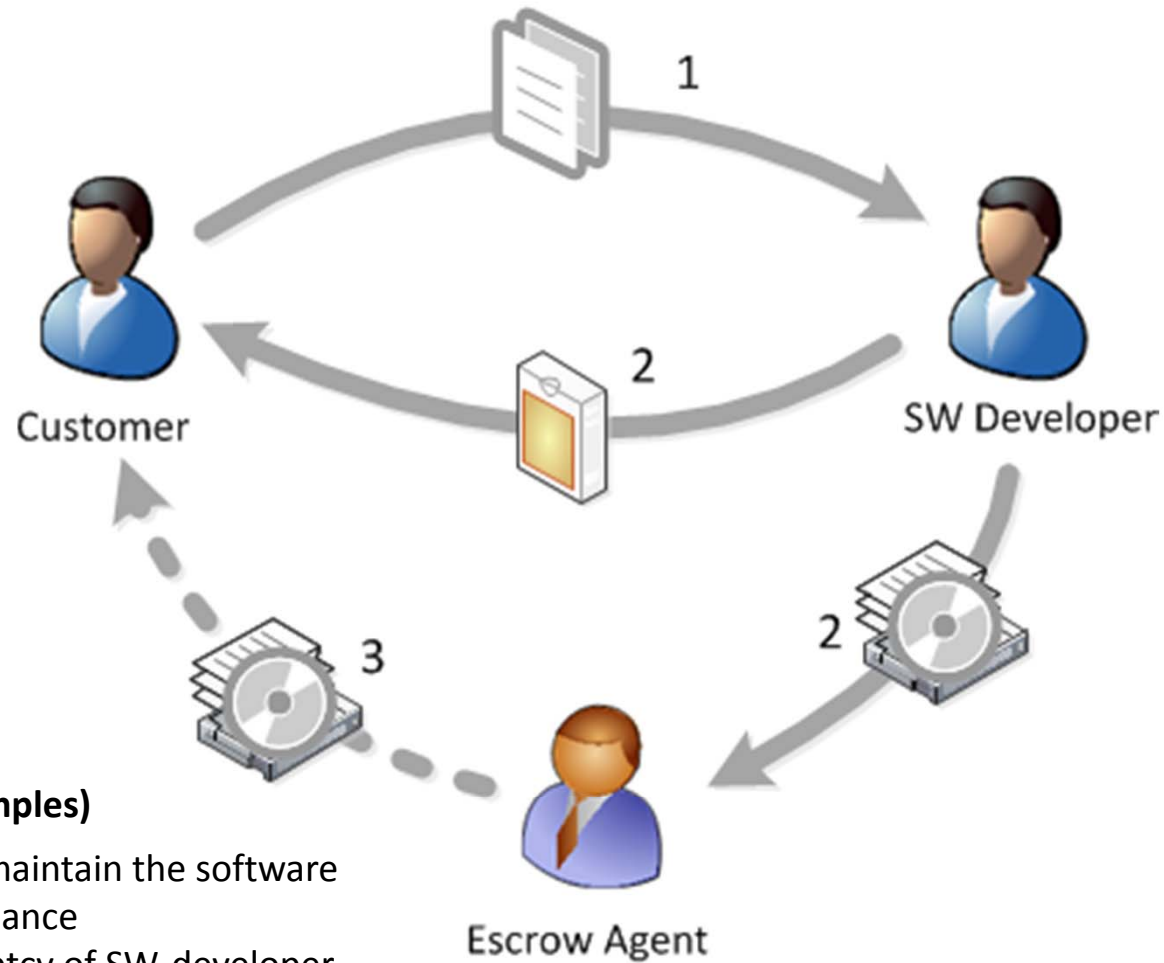
Source: Gartner

Risks of Outsourcing

- Lost of control
- Financial standing of the vendor
- Sale of the vendor
- Maintenance and adjustments of the system
- Business critical application



Software Escrow



Release events (examples)

- Failure to support/maintain the software
- Insufficient maintenance
- Insolvency/ bankruptcy of SW-developer

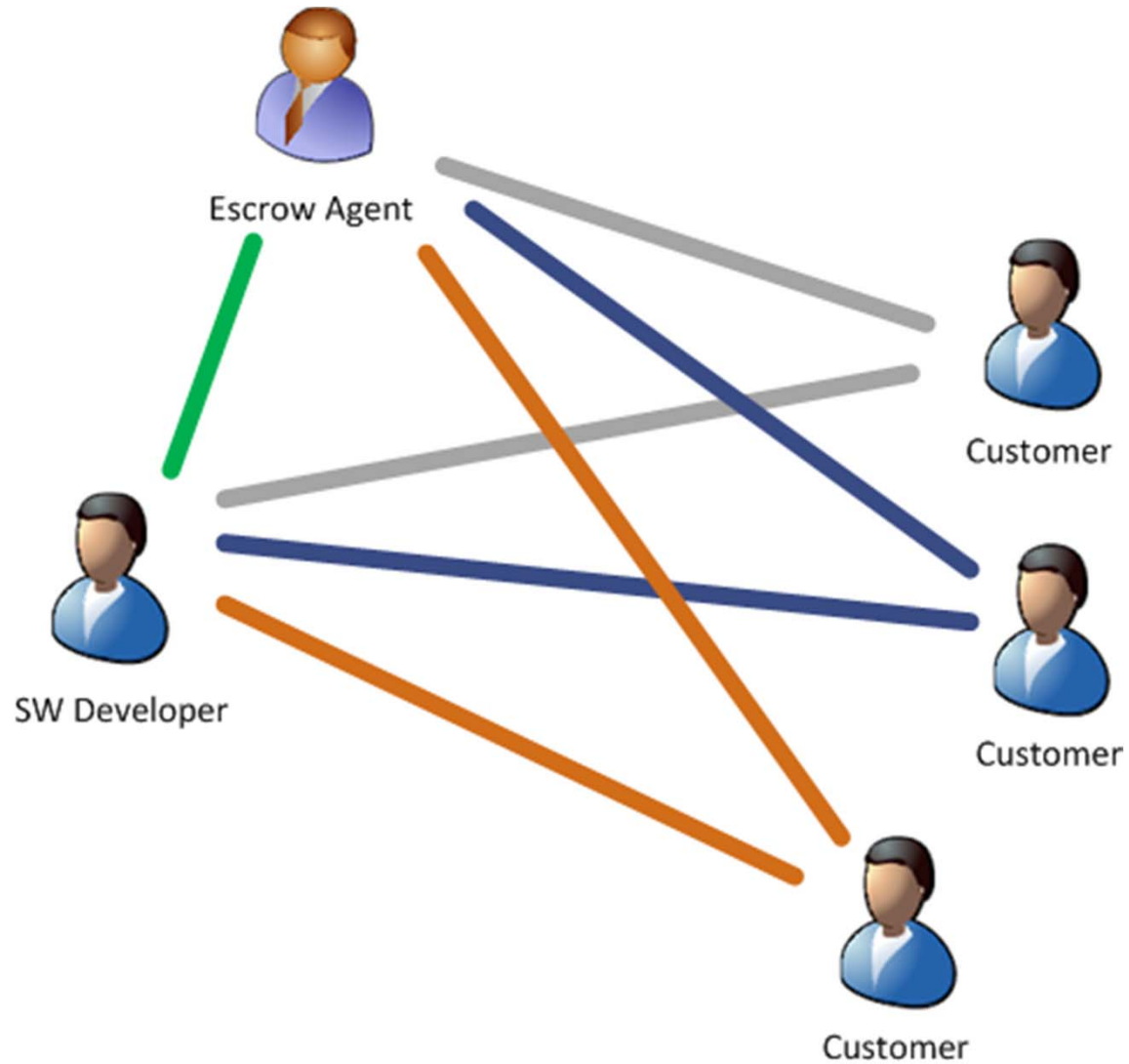
Escrow parties

- Customer
 - Use of the software
 - Protection of investment
- SW Developer
 - Provides object code to consumer
 - Commits source code to Escrow Agent
 - Obligations to the consumer as well as the Software Escrow Agent
- Escrow Agent
 - Verification of deposit material
 - Deposit of source code (Treuhand)
 - In case of agreed circumstances release of the source code to the consumer

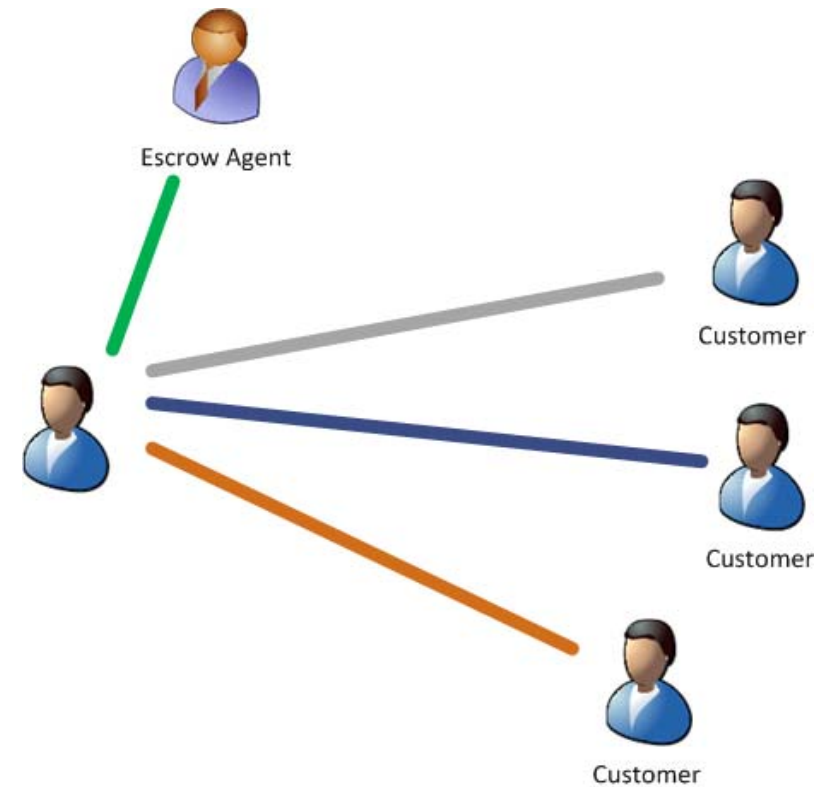
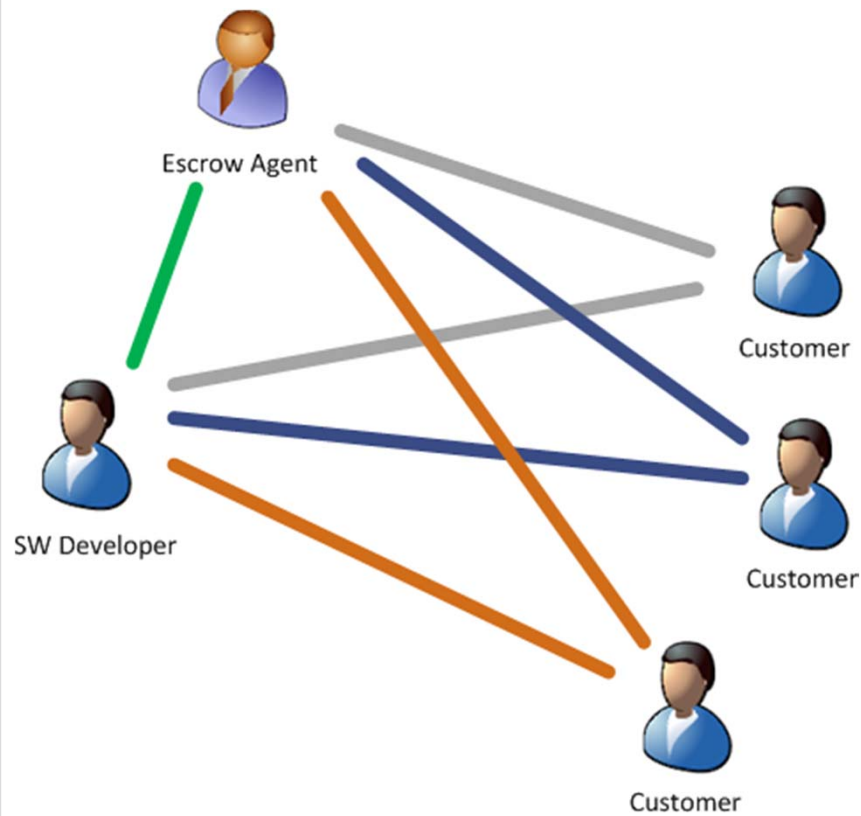
Motivation of parties

- Customer
 - Risk mitigation strategy
 - Maintenance and support
 - Protection of business
 - Hedging of investments
 - Verification of source code
- SW-Developer
 - Evidence of copyright
 - Warranty claim
 - Confidence-building measure
 - Payment

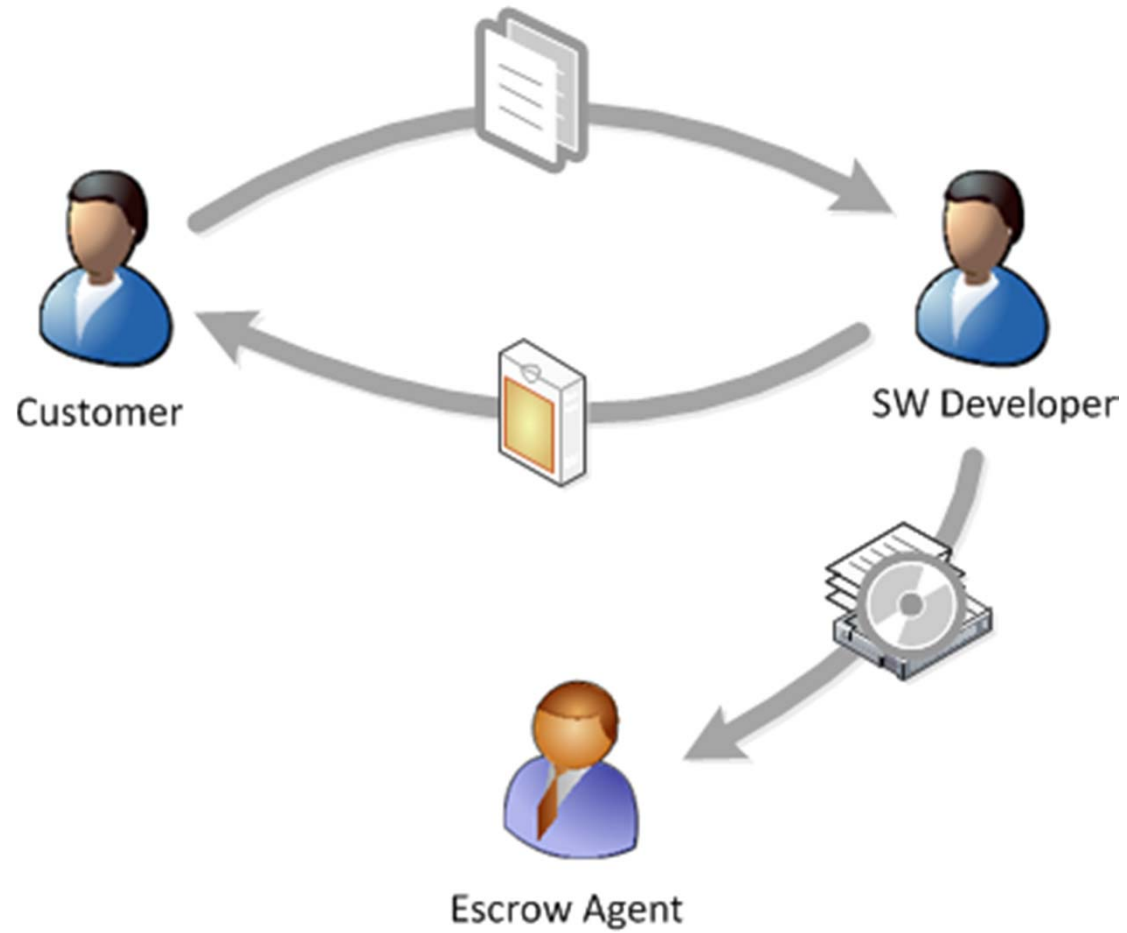
Escrow contract



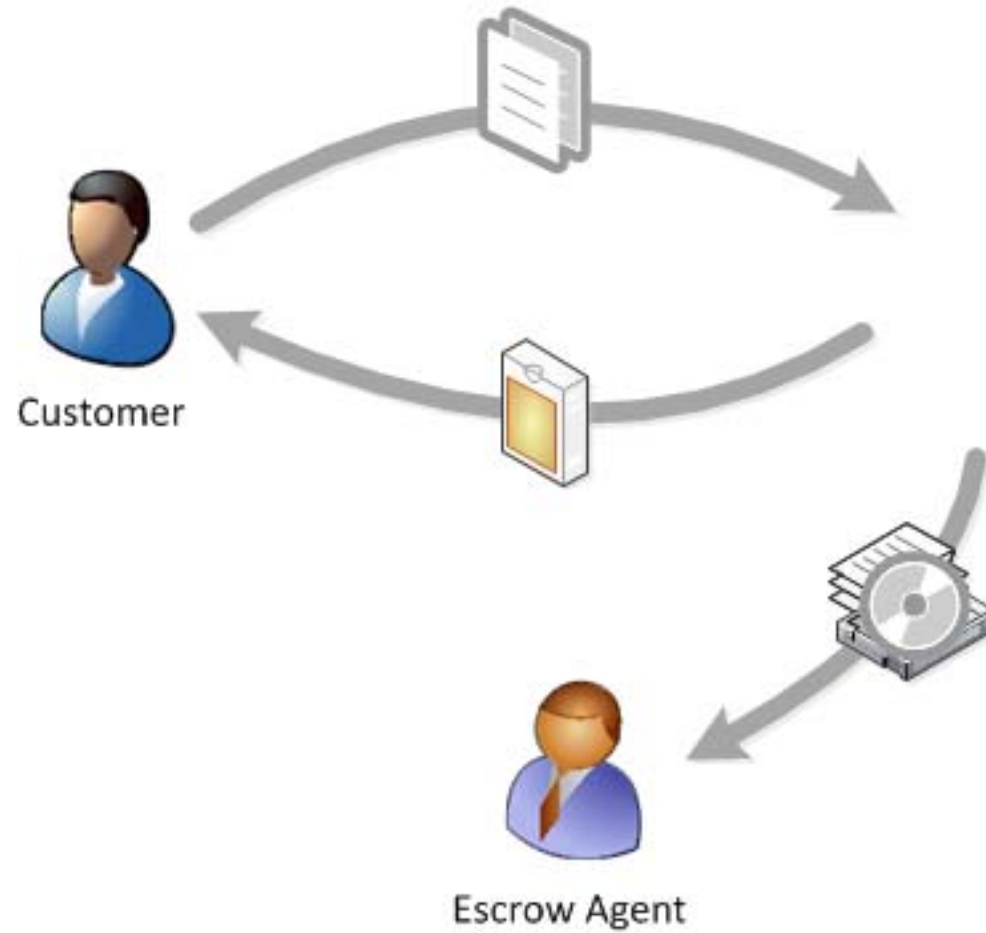
Unknown number of consumers



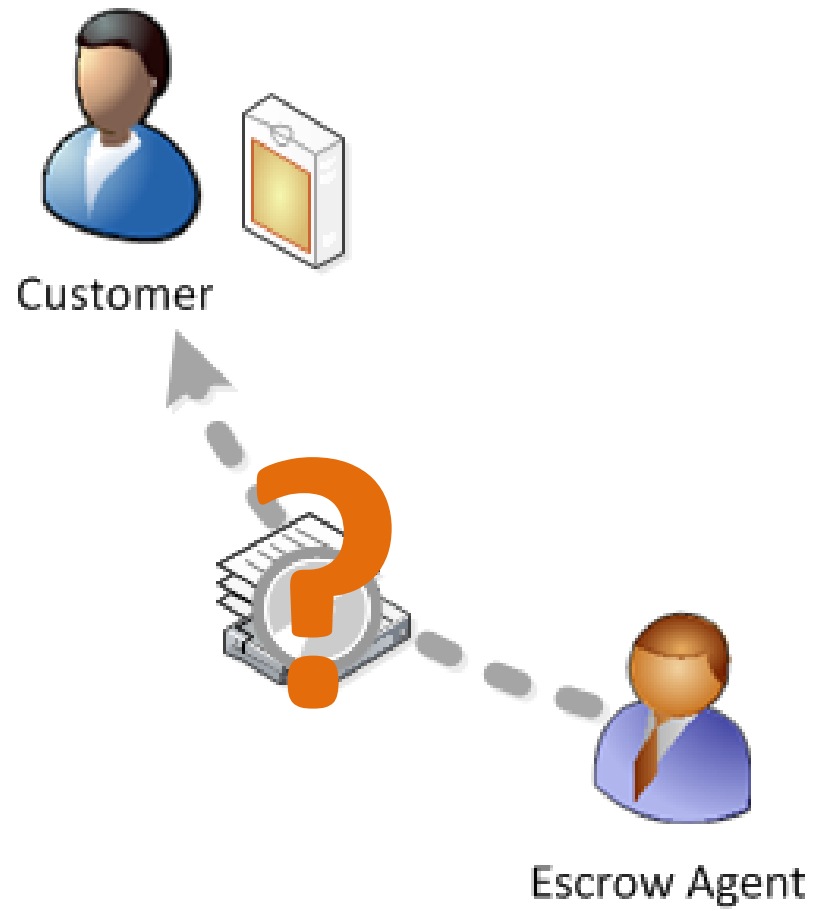
Escrow Case



Escrow Case



Escrow Case



Risks of Escrow

- Read error storage media
- Source code incomplete
- Build environment not deposited
- Configuration is not available
- Instructions missing

Risks of Escrow

- Test data missing
- Documentation insufficient
- Licences and rights not included (Development- & Build- Environment, Libraries)
- Deposited material is not up to date

```
def downloadURL(src, pageurl):
    print "downloading " + src
    time.sleep(random.random())
    try:
        downloadURL(src, ""+str(cardnumber)+"output")
    except urllib2.URLError, msg:
        print "ncfiles: urllib2 error (%s)" % msg
    except socket.error, (errno, strerror):
        print "ncfiles: Socket error (%s) for host %s (%s)" % (errno,
                                                                host,
                                                                strerror)

for h3 in page.findAll("h3"):
    value = (h3.contents[0])
    if value != "Afdeling":
        print >> txt, value
        import codecs
        f = codecs.open("alle.txt", "r", encoding="utf-8")
        text = f.read()
        f.close()
        # open the file again for writing
        f = codecs.open("alle.txt", "w", encoding="utf-8")
        f.write(value+"\n")
        # write the original contents
        f.write(text)
        f.close()

loadedURL[pageurl] = True
f.close()
f2.close()
system("mkdir "+str(cardnumber)+"products")
system("mv "+str(cardnumber)+"products/*.jpg "+str(cardnumber)+"products")
```

Requirements

- **Completeness**
 - Source code only part of a software
 - Without additional information almost impossible to understand, analyse, use and change the source code
- **Quality**
 - Up to date
 - Maintainability

Three phases of Escrow

- Phase I – Planning
 - Agreement on contract specifying measurements for Software Escrow
- Phase II – Execution
 - Deposit of material including iterations
- Phase III – Redeployment
 - Release of material if trigger event occurs

Phase I - Planning

- Selection of the escrow agent
 - Confidence of consumer and SW developer
 - Requirements for IT infrastructure
 - Data security requirements
 - Information-/notification obligation
- Alignment of the contracts
 - Inclusion of basic escrow provisions in the software licensing/maintenance contracts
- Select deposit material

Phase I - Planning

- Rights to the software artefacts
 - Rights of the escrow agent
 - Rights of the software developer
- Clarification of the legal consequences of the release
 - Determination of the rights transferred to the consumer
 - Obligations on confidentiality
- Setting up an Escrow contract

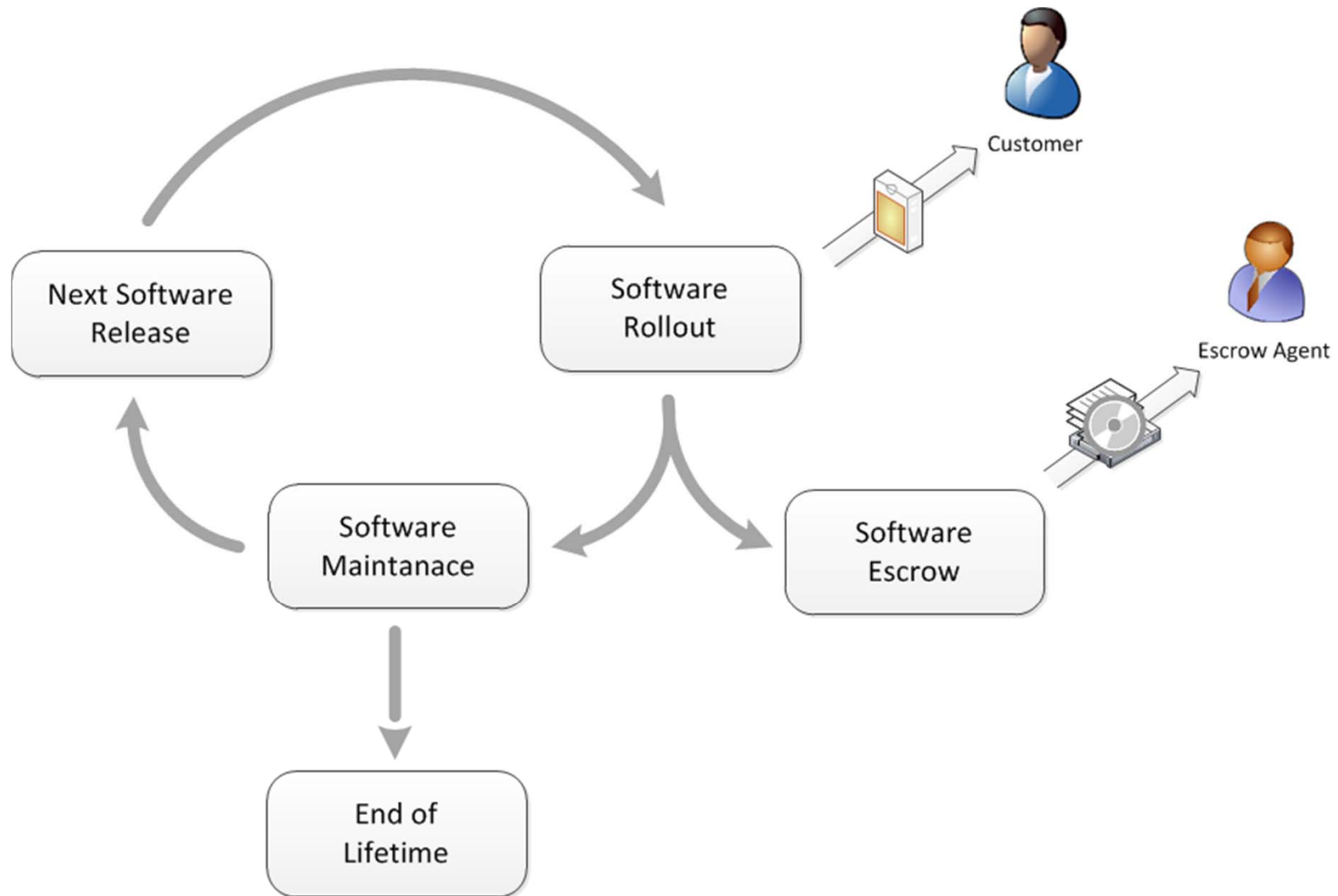
Phase II - Execution

- Deposit procedure
 - Submission deadlines
 - Existing verification procedures
 - Thresholds for successful deposit
 - Information-/notification obligations
- Verification of material
 - Technical Software Escrow Framework


Phase III - Redeployment

- Release of materials
 - Release events (insolvency, refusal to maintenance, ...)
 - Release process
- Information-/notification obligations

Software release cycles



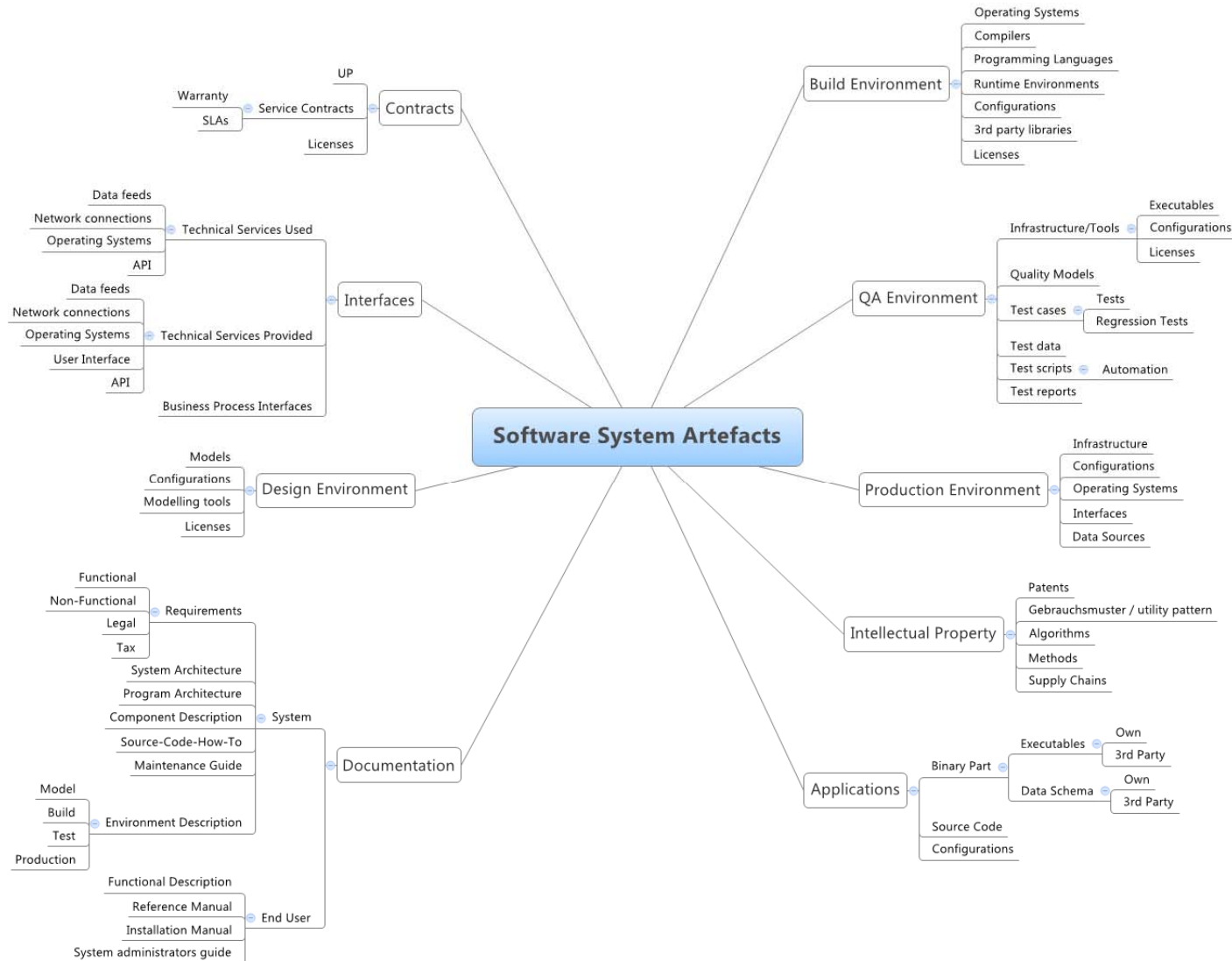
Escrow Audits

- Verification in literature
 - Standard or basic
Readable and virus free material, check list
 - Full verification 
compile of source code
 - Bespoke verification
negotiable (ensuring to meet specification, manual review)
- costs vs benefit

SW Quality



Artefacts of a software product



Escrow Audit

- Completeness of deposit material
 - Source code alone is not sufficient

- Quality measurements
 - Documentation
 - Understandably
 - Testability
 - Modifiability

Technical Framework for Audit

- Build environment
- Quality metric measurements
- Test environment
- Source code comments
- Specification verification
- System documentation
- Data sources
- External references
- Code obfuscation

Checks for Validation

- Build & Release
- Comments
 - Language detection
 - Comment text proof check
 - Comment ratio
- Complexity
 - Complexity calculation (McCabe, Halstead)
 - Comment-complexity ratio
- Nonstandard libraries
 - Hiding of source code
- Binary and external calls
 - Dependencies and hiding of source code
- License check

Escrow for Digital Preservation

- Deposit of software product for the long term
- Preservation strategy for third party services and software
- Maintainability
- Documentation

Thanks for your attention

Questions?