Content Management in the Next Generation Internet

Innovative Uses of the DOI System

Gregory L. Heileman
Digital Rights and Knowledge Engineering (DRAKE) Group
Department of Electrical & Computer Engineering
University of New Mexico
Albuquerque, NM

June 21, 2007

Introduction

- The International DOI Foundation distinguishes between different types of "identifiers":
  - Labels
  - Actionable labels
  - Implemented systems

  where the last item refers to a packaged system offering a label + tools + business model, an example of which is the DOI system.

Introduction

- The International DOI Foundation distinguishes between different types of "identifiers":
  - Labels
  - Actionable labels
  - Implemented systems

  where the last item refers to a packaged system offering a label + tools + business model, an example of which is the DOI system.
- N. Paskin and others have described some of the issues associated with the use of DOI in digital rights management (DRM) settings.

Introduction

- The International DOI Foundation distinguishes between different types of "identifiers":
  - Labels
  - Actionable labels
  - Implemented systems

  where the last item refers to a packaged system offering a label + tools + business model, an example of which is the DOI system.
- N. Paskin and others have described some of the issues associated with the use of DOI in digital rights management (DRM) settings.
- This talk will describe how current activities associated with the design of the next generation Internet may impact DRM infrastructures developed around the DOI system.

Introduction

- The International DOI Foundation distinguishes between different types of "identifiers":
  - Labels
  - Actionable labels
  - Implemented systems

  where the last item refers to a packaged system offering a label + tools + business model, an example of which is the DOI system.
- N. Paskin and others have described some of the issues associated with the use of DOI in digital rights management (DRM) settings.
- This talk will describe how current activities associated with the design of the next generation Internet may impact DRM infrastructures developed around the DOI system.
- We will also describe a demo that implements some interesting rights-related scenarios in a content management setting.

What lead to the emergence of DRM?

- The Internet provides tremendous opportunities for content-based commerce.

What lead to the emergence of DRM?

- The Internet provides tremendous opportunities for content-based commerce.

  Why?

  - frictionless distribution
  - worldwide reach

What lead to the emergence of DRM?

- The Internet provides tremendous opportunities for content-based commerce.

  Why?

  - frictionless distribution
  - worldwide reach

  Reality: Content industries are not making full use of the Internet.

What lead to the emergence of DRM?

- The Internet provides tremendous opportunities for content-based commerce.

  Why?
  - frictionless distribution
  - worldwide reach

  Reality: Content industries are not making full use of the Internet.

  Why?
  - frictionless distribution
  - worldwide reach

Introduction

Some have referred to what we are talking about as the digital dilemma:

Introduction

Some have referred to what we are talking about as the digital dilemma:

- Technological advances that have led to the creation of the Internet allow vast amounts of information to be readily accessed, and provide content owners a nearly frictionless distribution channel for their content.

Introduction

Some have referred to what we are talking about as the digital dilemma:

- Technological advances that have led to the creation of the Internet allow vast amounts of information to be readily accessed, and provide content owners a nearly frictionless distribution channel for their content.
- These same advances in technology have produced radical shifts in the ability to reproduce and illicitly distribute content.

Introduction

Some have referred to what we are talking about as the digital dilemma:

- Technological advances that have led to the creation of the Internet allow vast amounts of information to be readily accessed, and provide content owners a nearly frictionless distribution channel for their content.
- These same advances in technology have produced radical shifts in the ability to reproduce and illicitly distribute content.
    - The average high school student today can easily do the amount copying and the volume of distribution that would have required the significant investment of a major corporation only a few years ago.

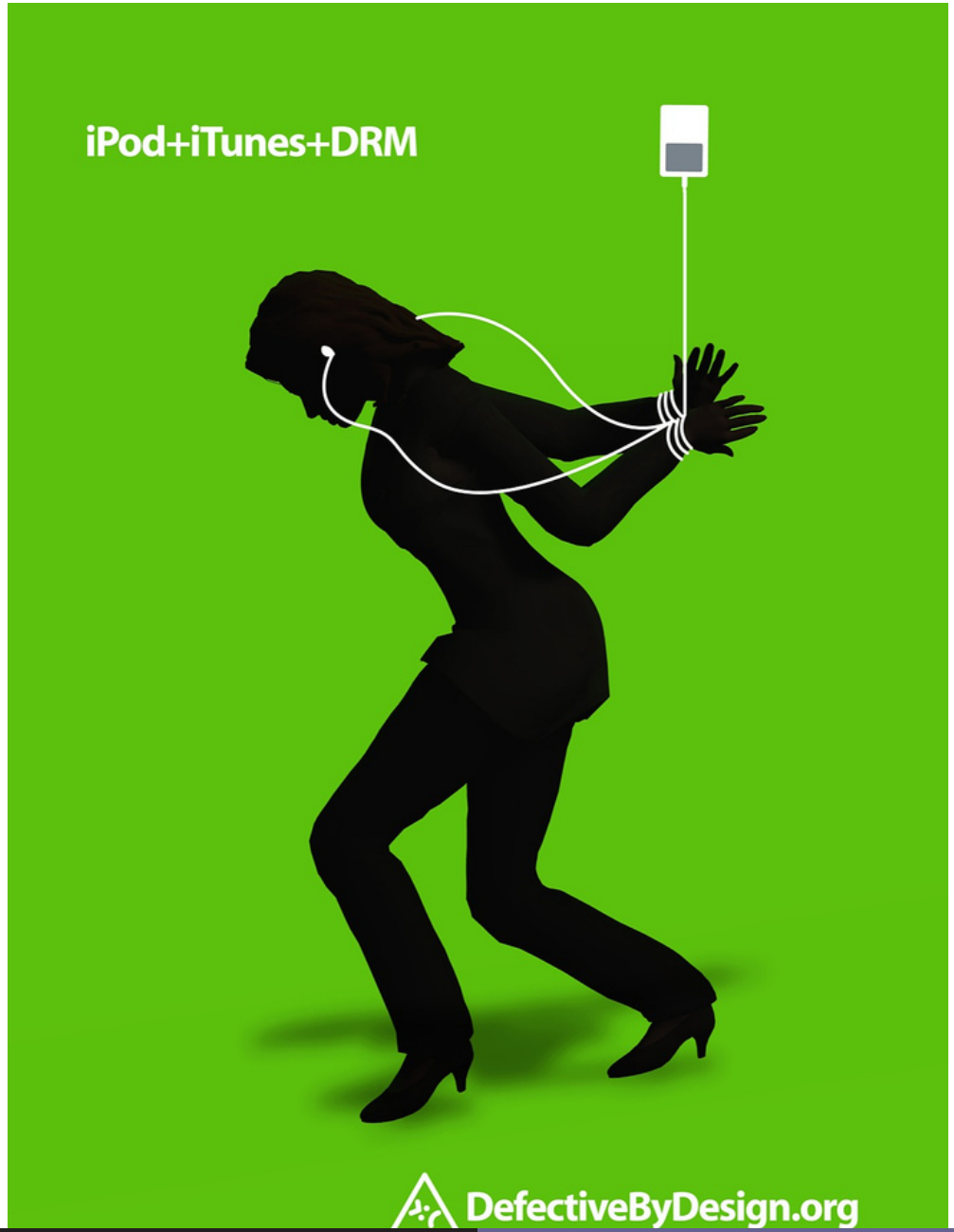
The Digital Dilemma – Industry View

The Digital Dilemma – Industry View

The Digital Dilemma – User View

The Digital Dilemma – User View

The Digital Dilemma – User View



The Digital Dilemma – User View

Content Management in the NGI

Goals:

- To provide the ability to manage/protect rights throughout the entire chain of the content distribution/usage pipeline.

Content Management in the NGI

Goals:

- To provide the ability to manage/protect rights throughout the entire chain of the content distribution/usage pipeline.
- To allow the creation of new business models related to digital content.

Content Management in the NGI

Goals:

- To provide the ability to manage/protect rights throughout the entire chain of the content distribution/usage pipeline.
- To allow the creation of new business models related to digital content.
- To add value to content.

Content Management in the NGI

Goals:

- To provide the ability to manage/protect rights throughout the entire chain of the content distribution/usage pipeline.
- To allow the creation of new business models related to digital content.
- To add value to content.
- To allow content distributors to learn more about their customers.

Content Management in the NGI

- N. Paskin has described the importance that naming and metadata will play in the management of intellectual property in digital media.

Content Management in the NGI

- N. Paskin has described the importance that naming and metadata will play in the management of intellectual property in digital media.
- In particular, interoperability within a particular domain cannot be achieved without addressing these matters.

Content Management in the NGI

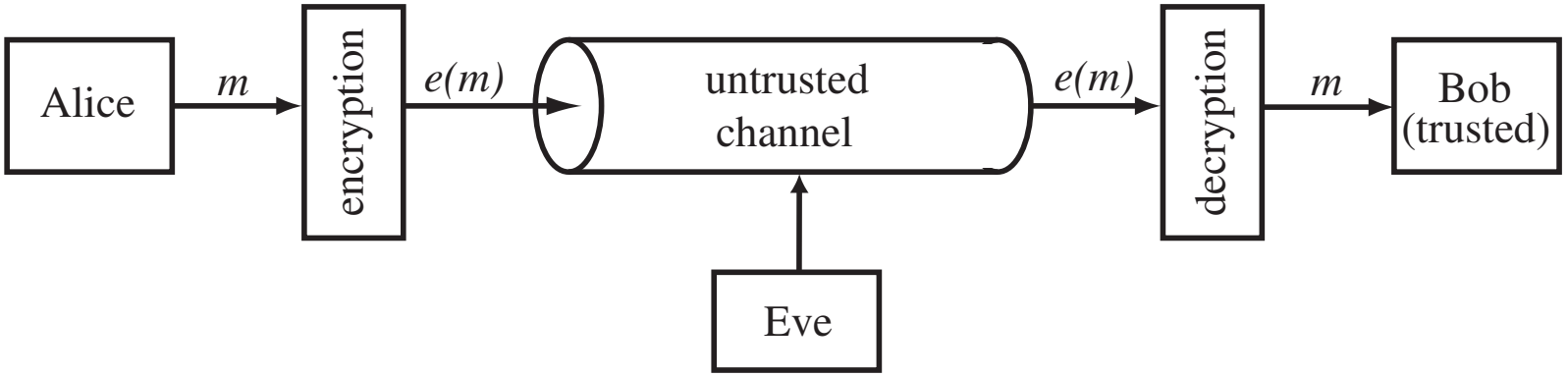
- N. Paskin has described the importance that naming and metadata will play in the management of intellectual property in digital media.
- In particular, interoperability within a particular domain cannot be achieved without addressing these matters.
- However, these address only one level of interoperability associated with whether or not objects know each other (naming), and whether or not they can talk to each other (metadata/ontologies).

Content Management in the NGI

- N. Paskin has described the importance that naming and metadata will play in the management of intellectual property in digital media.
- In particular, interoperability within a particular domain cannot be achieved without addressing these matters.
- However, these address only one level of interoperability associated with whether or not objects know each other (naming), and whether or not they can talk to each other (metadata/ontologies).
- DRM poses further challenges that come about due to security-related issues, and these introduce additional barriers to interoperability.

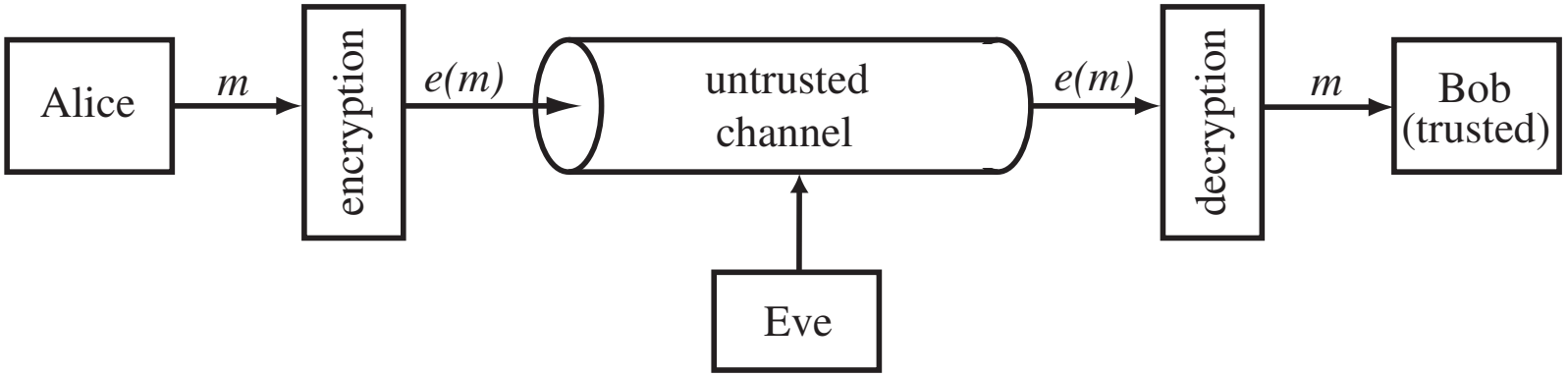
Cryptography Scenario

Consider the typical cryptography scenario:



Cryptography Scenario

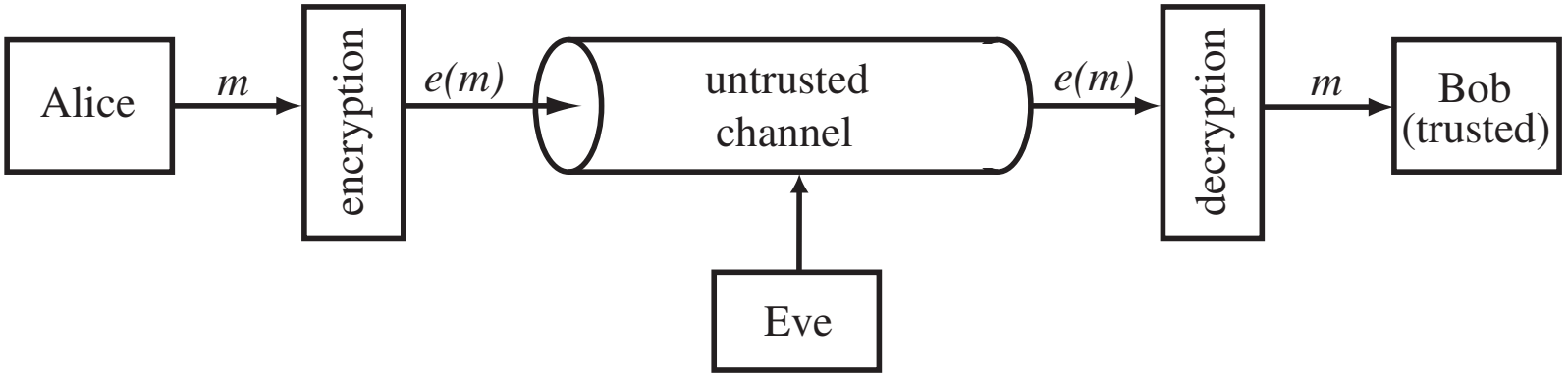
Consider the typical cryptography scenario:



- Goal: To communicate a message from Alice to Bob through a untrusted channel in such a way that Bob can read the message, but not Eve.

Cryptography Scenario

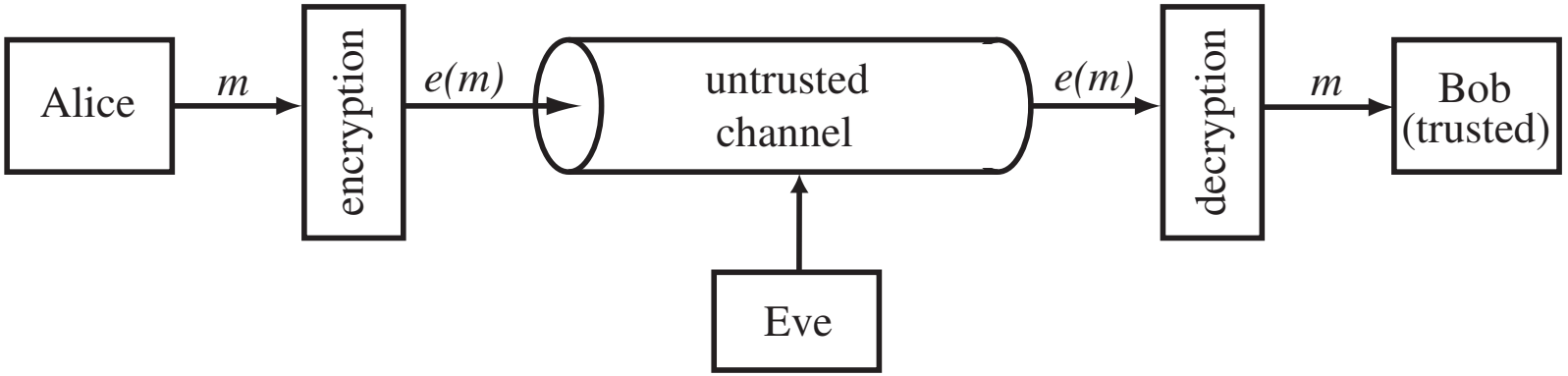
Consider the typical cryptography scenario:



- Goal: To communicate a message from Alice to Bob through a untrusted channel in such a way that Bob can read the message, but not Eve.
- Bob has the message in "cleartext", but Alice trusts him to handle it appropriately.

Cryptography Scenario

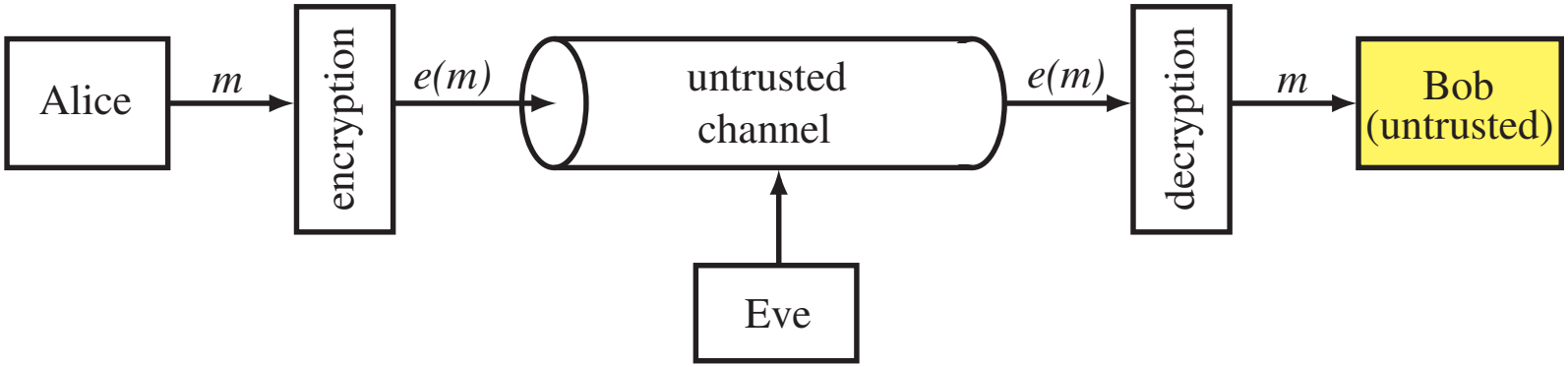
Consider the typical cryptography scenario:



- Goal: To communicate a message from Alice to Bob through a untrusted channel in such a way that Bob can read the message, but not Eve.
- Bob has the message in "cleartext", but Alice trusts him to handle it appropriately.
- The are well-defined mathematical notions regarding the capabilities of cryptosystems (e.g., unconditionally secure, computational secure, etc.).

DRM Scenario

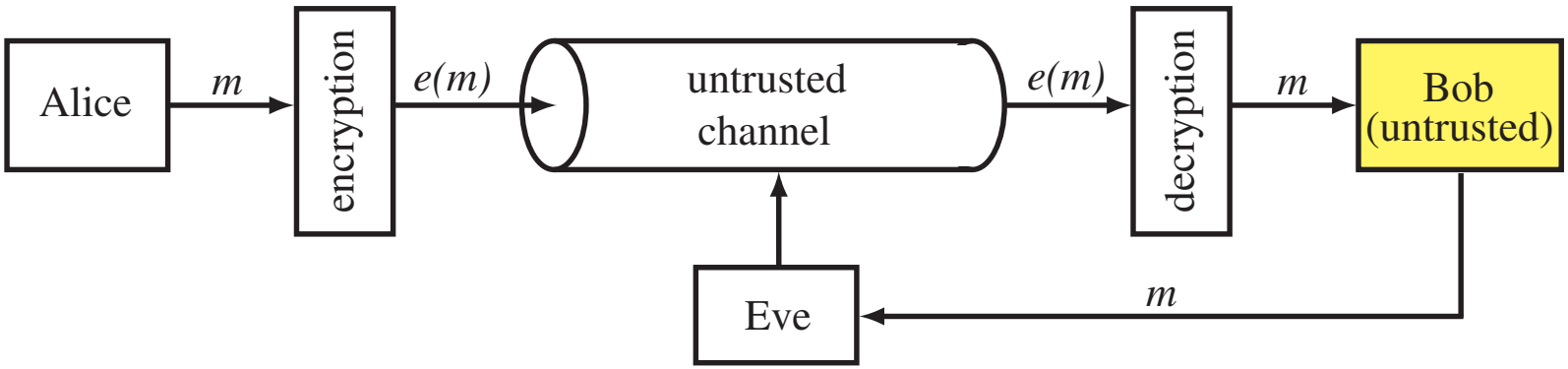
Now consider the typical DRM scenario:



- Goal: To provide content from Alice to Bob in such a way that it cannot be accessed by Eve, *and usage of the content by Bob is "managed"*.

DRM Scenario

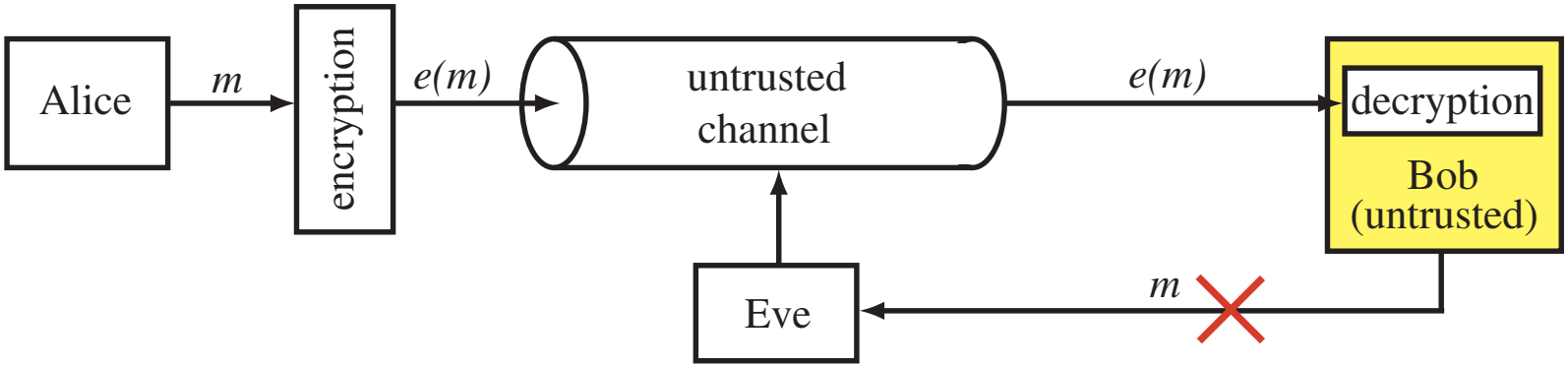
Now consider the typical DRM scenario:



- Goal: To provide content from Alice to Bob in such a way that it cannot be accessed by Eve, *and usage of the content by Bob is "managed"*.
- It is assumed that Bob cannot be trusted. Thus, he may provide the content to Eve.

DRM Scenario

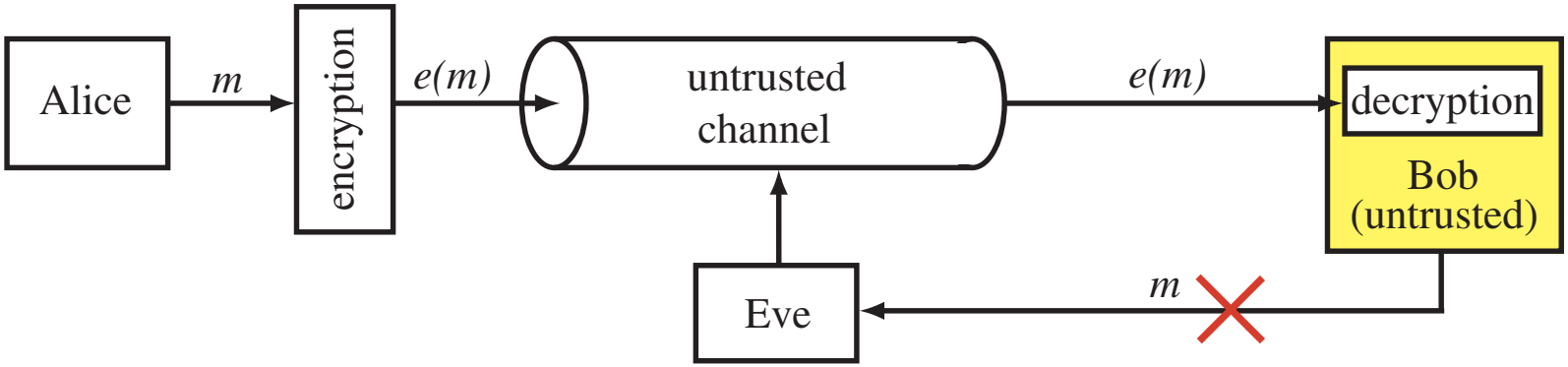
Now consider the typical DRM scenario:



- Goal: To provide content from Alice to Bob in such a way that it cannot be accessed by Eve, *and usage of the content by Bob is "managed"*.
- It is assumed that Bob cannot be trusted. Thus, he may provide the content to Eve.
- How can we prevent Bob from doing this?

DRM Scenario

Now consider the typical DRM scenario:



- Goal: To provide content from Alice to Bob in such a way that it cannot be accessed by Eve, *and usage of the content by Bob is "managed"*.
- It is assumed that Bob cannot be trusted. Thus, he may provide the content to Eve.
- How can we prevent Bob from doing this?
  - Secure container on Bob's machine.
  - Properly "motivate" Bob to play by the rules.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.
  Pros: effective in certain markets; established mechanisms. A controls problem?

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.
  Pros: effective in certain markets; established mechanisms. A controls problem?
  Cons: jurisdiction/credibility; Bob does not like to be threatened.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.
  Pros: effective in certain markets; established mechanisms. A controls problem?
  Cons: jurisdiction/credibility; Bob does not like to be threatened.
- Incentives– provide financial or other incentives to Bob for "handling" the content appropriately.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.
  Pros: effective in certain markets; established mechanisms. A controls problem?
  Cons: jurisdiction/credibility; Bob does not like to be threatened.
- Incentives– provide financial or other incentives to Bob for "handling" the content appropriately.
  Pros: align interests of Alice with those of Bob, creates a win-win situation. We're looking at analyzing this using game theory.

DRM Scenario

Ways of preventing Bob from mishandling content:

- Secure containers – software/hardware on Bob's machine that prevents him from accessing the content in its decrypted form – thus he never actually has the opportunity to handle m.
  Pros: A technological solution.
  Cons: No mathematical formulations of security; juicy target; resistance to installation; may violate certain IPRs.
- Threats/inconveniences – through legal or other means, make Bob afraid to "mishandle" the content.
  Pros: effective in certain markets; established mechanisms. A controls problem?
  Cons: jurisdiction/credibility; Bob does not like to be threatened.
- Incentives– provide financial or other incentives to Bob for "handling" the content appropriately.
  Pros: align interests of Alice with those of Bob, creates a win-win situation. We're looking at analyzing this using game theory.
  Cons: due to cheating, hard to create win-win situations.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.
- The Apple iTunes/FairPlay/iPod framework and the Microsoft Zune framework are prime examples.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.
- The Apple iTunes/FairPlay/iPod framework and the Microsoft Zune framework are prime examples. These systems do not separate rights from devices. The focus of some of our current work – TNA architecture separates logical functionality from local implementation.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.
- The Apple iTunes/FairPlay/iPod framework and the Microsoft Zune framework are prime examples. These systems do not separate rights from devices. The focus of some of our current work – TNA architecture separates logical functionality from local implementation.
- What we want are reusable and expandable set of services and functions that are commonly needed by many applications to function well in a networked environment.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.
- The Apple iTunes/FairPlay/iPod framework and the Microsoft Zune framework are prime examples. These systems do not separate rights from devices. The focus of some of our current work – TNA architecture separates logical functionality from local implementation.
- What we want are reusable and expandable set of services and functions that are commonly needed by many applications to function well in a networked environment.
- This is in fact the function of middleware — for DRM to become more prevalent, and deal effectively with issues such as interoperability, it should be configured as middleware services.

DRM Architectures

- Current DRM infrastructures – single vendor, monolithic, inflexible, closed.
- The Apple iTunes/FairPlay/iPod framework and the Microsoft Zune framework are prime examples. These systems do not separate rights from devices. The focus of some of our current work – TNA architecture separates logical functionality from local implementation.
- What we want are reusable and expandable set of services and functions that are commonly needed by many applications to function well in a networked environment.
- This is in fact the function of middleware — for DRM to become more prevalent, and deal effectively with issues such as interoperability, it should be configured as middleware services.
- We will see that the structure of rights expression languages has a profound influence on the ability to do this.

Current Internet

- One of the most important design considerations associated with the current Internet are the so-called end-to-end arguments.

Current Internet

- One of the most important design considerations associated with the current Internet are the so-called end-to-end arguments.
  - Most of the intelligence is located at the end points, with a dumb core.
  - This has been changing over time – but it's been difficult because we're building on top of protocols that were not meant to support these notions.

Current Internet

- One of the most important design considerations associated with the current Internet are the so-called end-to-end arguments.
  - Most of the intelligence is located at the end points, with a dumb core.
  - This has been changing over time – but it's been difficult because we're building on top of protocols that were not meant to support these notions.
- In the future more intelligence will move into the core, perhaps including the ability to identify individual content objects within the network.

Next Generation Internet

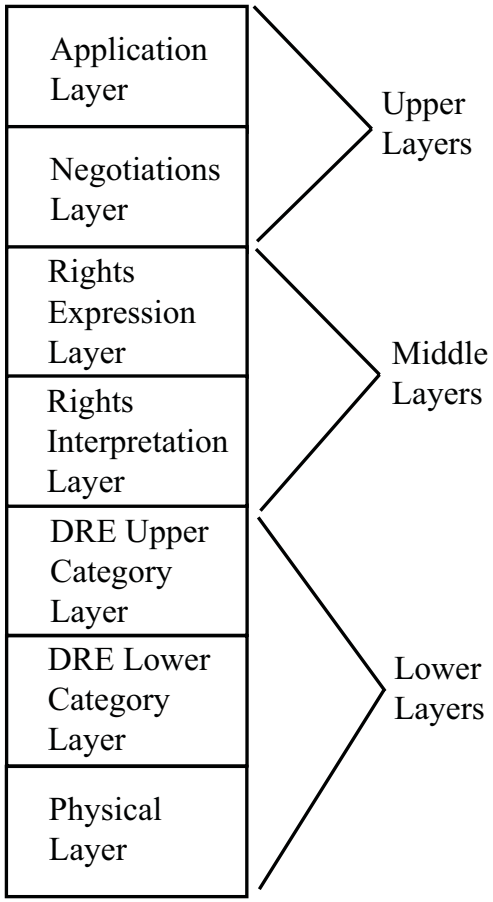
- The next generation Internet:
  - More intelligence in the network.
  - New kinds of naming and addressing.
  - Designing for disconnections.

Next Generation Internet

- The next generation Internet:
  - More intelligence in the network.
  - New kinds of naming and addressing.
  - Designing for disconnections.
- TNA Architecture – we no longer depend upon an end point and the node that hosts it, but instead on a logical abstraction of the functionality that needs to be provided.

DRM Architectures

We have proposed an abstract layered architectural framework (similar to the OSI layers) for dealing with the problem of allowing multiple vendors to participate in the pipeline:



Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.
- Involves specifying rights and rights-related scenarios in machine-readable and machine-actionable form.

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.
- Involves specifying rights and rights-related scenarios in machine-readable and machine-actionable form.
- A machine-actionable REL must use very precise language in order to guarantee compliance with a machine-readable license

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.
- Involves specifying rights and rights-related scenarios in machine-readable and machine-actionable form.
- A machine-actionable REL must use very precise language in order to guarantee compliance with a machine-readable license $\Rightarrow$ makes it difficult to support social or legal issues such as "fair-use" which tend to have "fuzzy" interpretations.

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.
- Involves specifying rights and rights-related scenarios in machine-readable and machine-actionable form.
- A machine-actionable REL must use very precise language in order to guarantee compliance with a machine-readable license $\Rightarrow$ makes it difficult to support social or legal issues such as "fair-use" which tend to have "fuzzy" interpretations.
- An REL generally must exist within the context of a larger system that contains the business environment the REL supports, the delivery network, and the rendering platform.

Rights Expression Languages

- The first rights expression languages (RELs) were developed in the late 1990's, and only now are they starting to be deployed.
- RELs are used to: express copyright, express contract or license agreements, and control access/usage of content.
- Involves specifying rights and rights-related scenarios in machine-readable and machine-actionable form.
- A machine-actionable REL must use very precise language in order to guarantee compliance with a machine-readable license $\Rightarrow$ makes it difficult to support social or legal issues such as "fair-use" which tend to have "fuzzy" interpretations.
- An REL generally must exist within the context of a larger system that contains the business environment the REL supports, the delivery network, and the rendering platform.
- Various RELs have been proposed, some of them aimed at solving rights-related problems within a particular content industry.

Example RELs

- Open Digital Rights Language (ODRL) – Designed by IPR systems (Australia) as an open standard. Now a cooperative project with more than a dozen participating organizations. Consists of an expression language and a data dictionary. XML based.

Example RELs

- Open Digital Rights Language (ODRL) – Designed by IPR systems (Australia) as an open standard. Now a cooperative project with more than a dozen participating organizations. Consists of an expression language and a data dictionary. XML based.
- MPEG-21/5 – Accepted as ISO 21000. A REL only, provides a structure for a data dictionary for the REL, but a complete data dictionary has not been developed. XML based.

Example RELs

- Open Digital Rights Language (ODRL) – Designed by IPR systems (Australia) as an open standard. Now a cooperative project with more than a dozen participating organizations. Consists of an expression language and a data dictionary. XML based.
- MPEG-21/5 – Accepted as ISO 21000. A REL only, provides a structure for a data dictionary for the REL, but a complete data dictionary has not been developed. XML based.
- CreativeCommons (CC) – Provides an exprssion of rights for open access web resources, including HTML documents, RSS feeds, and digital audio files. License is machine-readable, but there is no machine-actionable control over use of the content.

Example RELs

- Open Digital Rights Language (ODRL) – Designed by IPR systems (Australia) as an open standard. Now a cooperative project with more than a dozen participating organizations. Consists of an expression language and a data dictionary. XML based.
- MPEG-21/5 – Accepted as ISO 21000. A REL only, provides a structure for a data dictionary for the REL, but a complete data dictionary has not been developed. XML based.
- CreativeCommons (CC) – Provides an exprssion of rights for open access web resources, including HTML documents, RSS feeds, and digital audio files. License is machine-readable, but there is no machine-actionable control over use of the content.
- METSRights (METSR) – Mainly used in academic and library-based environments providing content for education and research purposes, many of these are archival in nature. XML based.

Abstract REL – Rights

M. Stefik identified the following fundamental rights associated with content:

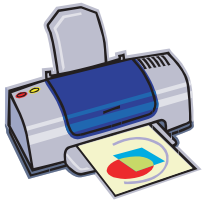
Render Rights:

Abstract REL – Rights

M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print

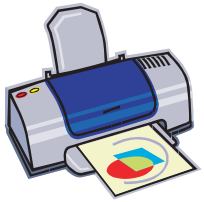


Abstract REL – Rights

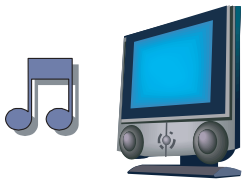
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View

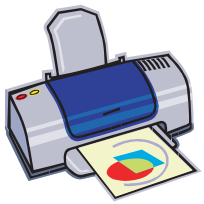


Abstract REL – Rights

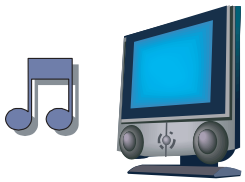
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



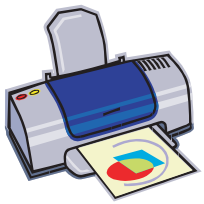
- Export

Abstract REL – Rights

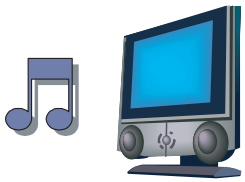
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

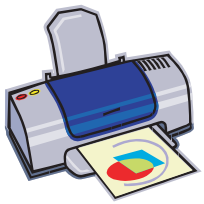
Transport Rights:

Abstract REL – Rights

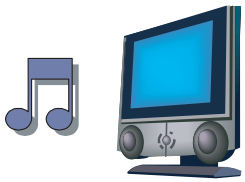
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

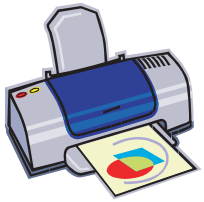
- Copy

Abstract REL – Rights

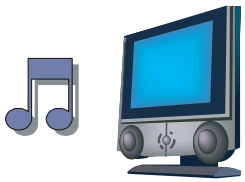
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

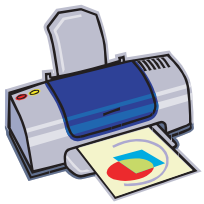
- Copy

Abstract REL – Rights

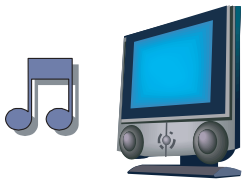
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

- Copy

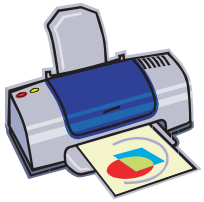
- Move/Transfer

Abstract REL – Rights

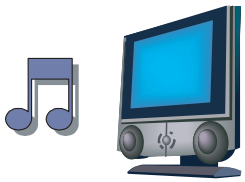
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

- Copy

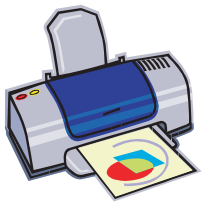
- Move/Transfer

Abstract REL – Rights

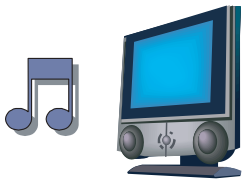
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

- Copy

- Move/Transfer

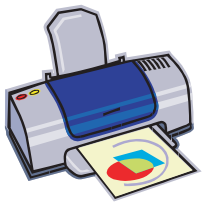
- Loan

Abstract REL – Rights

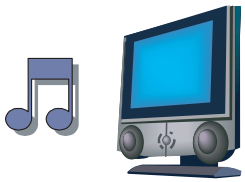
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



- Export

Transport Rights:

- Copy

- Move/Transfer

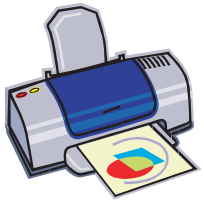
- Loan

Abstract REL – Rights

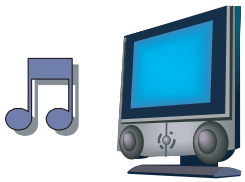
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



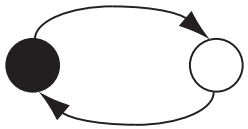
- Export

Transport Rights:

- Copy

- Move/Transfer

- Loan

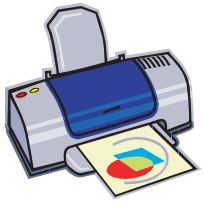


Abstract REL – Rights

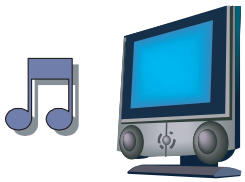
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



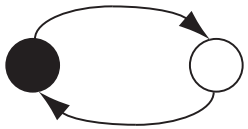
- Export

Transport Rights:

- Copy

- Move/Transfer

- Loan



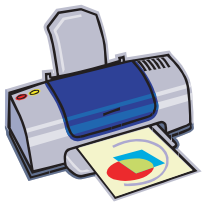
Derivative Work Rights:

Abstract REL – Rights

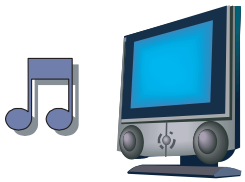
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



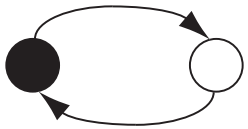
- Export

Transport Rights:

- Copy

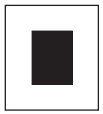
- Move/Transfer

- Loan



Derivative Work Rights:

- Extract

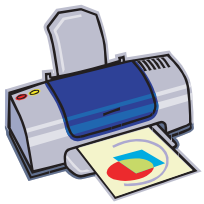


Abstract REL – Rights

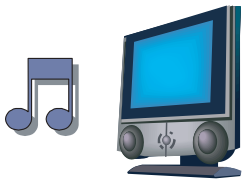
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



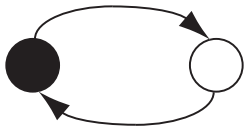
- Export

Transport Rights:

- Copy

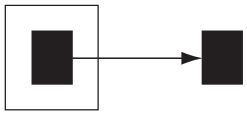
- Move/Transfer

- Loan



Derivative Work Rights:

- Extract

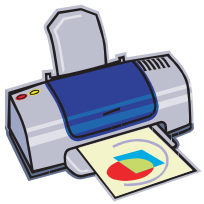


Abstract REL – Rights

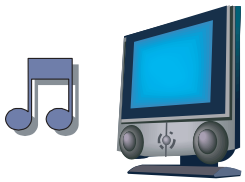
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



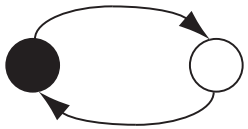
- Export

Transport Rights:

- Copy

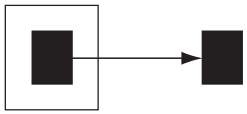
- Move/Transfer

- Loan

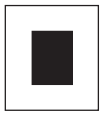


Derivative Work Rights:

- Extract



- Edit

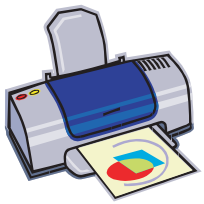


Abstract REL – Rights

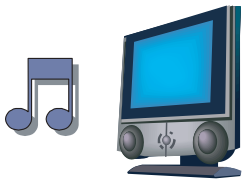
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



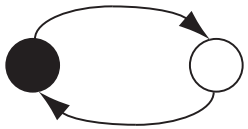
- Export

Transport Rights:

- Copy

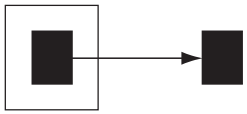
- Move/Transfer

- Loan

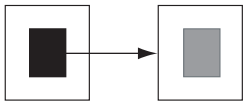


Derivative Work Rights:

- Extract



- Edit

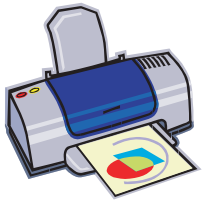


Abstract REL – Rights

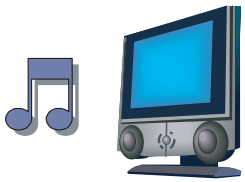
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



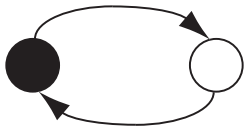
- Export

Transport Rights:

- Copy

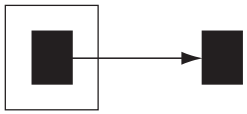
- Move/Transfer

- Loan

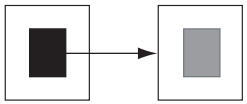


Derivative Work Rights:

- Extract



- Edit



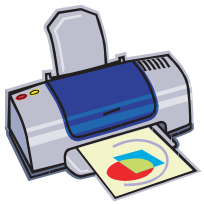
- Embed

Abstract REL – Rights

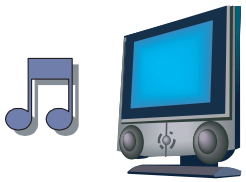
M. Stefik identified the following fundamental rights associated with content:

Render Rights:

- Print



- Play/View



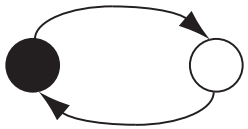
- Export

Transport Rights:

- Copy

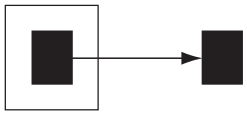
- Move/Transfer

- Loan

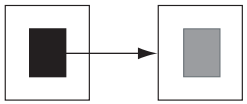


Derivative Work Rights:

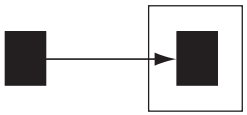
- Extract



- Edit



- Embed



REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.

REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.
- It is often noted that DRM settings can become quite complex, and an REL is often faulted if it is not able to handle a particular DRM scenario.

REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.
- It is often noted that DRM settings can become quite complex, and an REL is often faulted if it is not able to handle a particular DRM scenario.
- In response, RELs have become highly complex.

REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.
- It is often noted that DRM settings can become quite complex, and an REL is often faulted if it is not able to handle a particular DRM scenario.
- In response, RELs have become highly complex.
- E.g., XrML and ODRL are much more than "rights expression" languages.
  - License management is in the core of XrML.

REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.
- It is often noted that DRM settings can become quite complex, and an REL is often faulted if it is not able to handle a particular DRM scenario.
- In response, RELs have become highly complex.
- E.g., XrML and ODRL are much more than "rights expression" languages.
  - License management is in the core of XrML.
  - Tracking is part of the ODRL rights expression model.

REL Background

- As we've seen a number of RELs have been proposed with the general intention of supporting DRM activities across networked environments, where interoperability becomes one of the most important considerations.
- It is often noted that DRM settings can become quite complex, and an REL is often faulted if it is not able to handle a particular DRM scenario.
- In response, RELs have become highly complex.
- E.g., XrML and ODRL are much more than "rights expression" languages.
  - License management is in the core of XrML.
  - Tracking is part of the ODRL rights expression model.
- It seems more appropriate to call them "DRM" languages.

An Analogy

Database	DRM
relational model	rights model

An Analogy

Database	DRM
relational model	rights model
tuple calculus	core REL (rights calculus)

An Analogy

Database	DRM
relational model	rights model
tuple calculus	core REL (rights calculus)
SQL	rights language

An Analogy

Database	DRM
relational model	rights model
tuple calculus	core REL (rights calculus)
SQL	rights language
application	application

An Analogy

Database	DRM
relational model	rights model
tuple calculus	core REL (rights calculus)
SQL	rights language
application	application

- What would database management systems look like if they had evolved like DRM systems?

Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.

Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.
  - Commercial DRM applications have largely ignored the use of standardized RELs.

Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.
  - Commercial DRM applications have largely ignored the use of standardized RELs.
- With current RELs there is no clean separation between rights expression and DRM services — this makes it difficult to compose different DRM services.

Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.
  - Commercial DRM applications have largely ignored the use of standardized RELs.
- With current RELs there is no clean separation between rights expression and DRM services — this makes it difficult to compose different DRM services.
  - A clean separation of rights from services is the key to creating interoperable DRM systems.

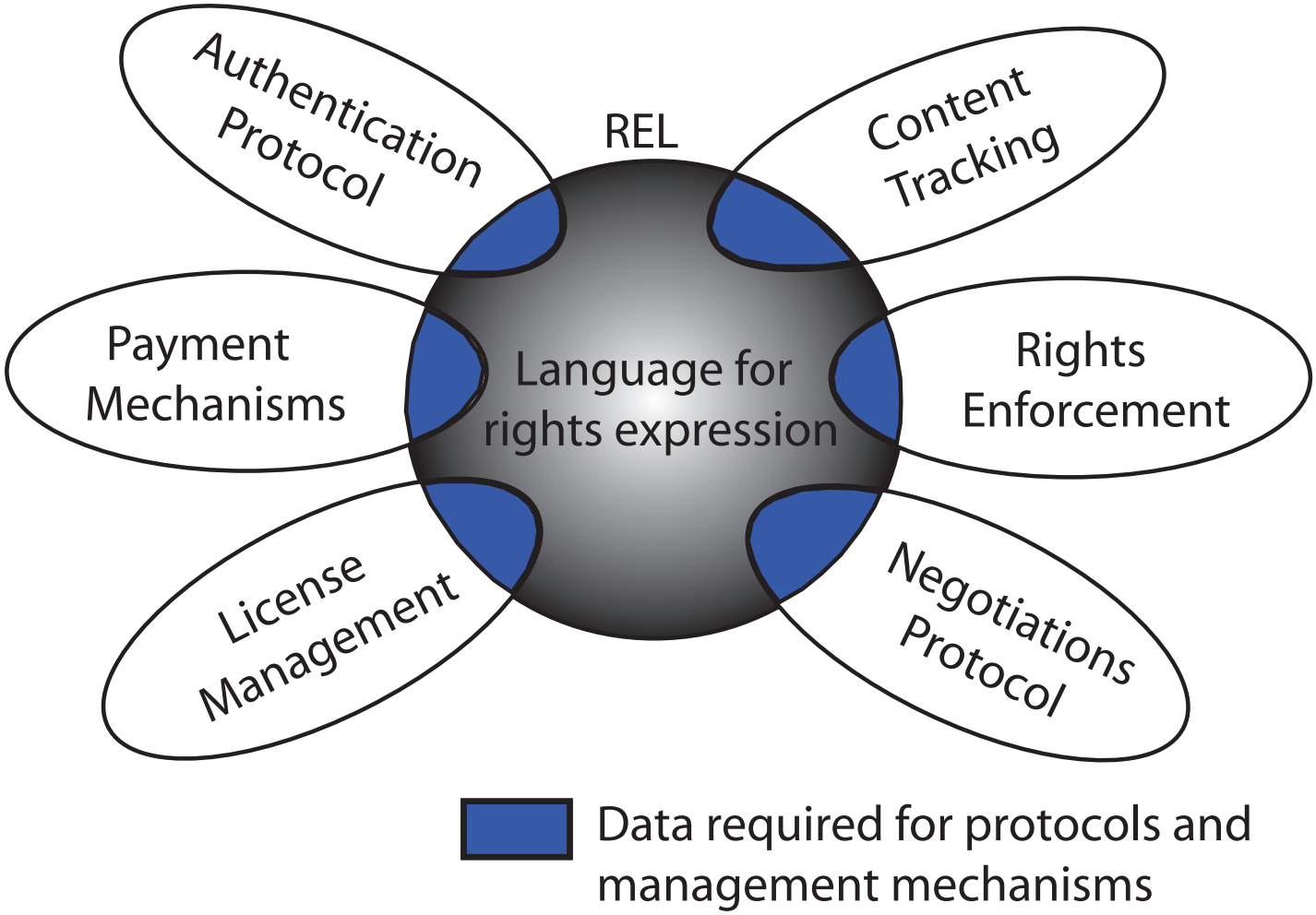
Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.
  - Commercial DRM applications have largely ignored the use of standardized RELs.
- With current RELs there is no clean separation between rights expression and DRM services — this makes it difficult to compose different DRM services.
  - A clean separation of rights from services is the key to creating interoperable DRM systems.
  - A refactoring of RELs is warranted.

Problems

- In order to use an REL, you are forced to include far more than may be necessary for your purposes.
  - Commercial DRM applications have largely ignored the use of standardized RELs.
- With current RELs there is no clean separation between rights expression and DRM services — this makes it difficult to compose different DRM services.
  - A clean separation of rights from services is the key to creating interoperable DRM systems.
  - A refactoring of RELs is warranted.
  - The middleware of yesterday can become the fundamental network infrastructure of tomorrow, e.g., DNS, PKI

DRM Services – Current RELs



Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

1. **The core REL should only contain the rights model.**

Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

1. **The core REL should only contain the rights model.**
2. **The core REL should be stateless.**

Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

1. **The core REL should only contain the rights model.**
2. **The core REL should be stateless.**
3. **The core REL should be language neutral.**

Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

1. **The core REL should only contain the rights model.**
2. **The core REL should be stateless.**
3. **The core REL should be language neutral.**
4. **REL primitives, and DRM services in general, should refer generically to the services they use.**

Recommendations

Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

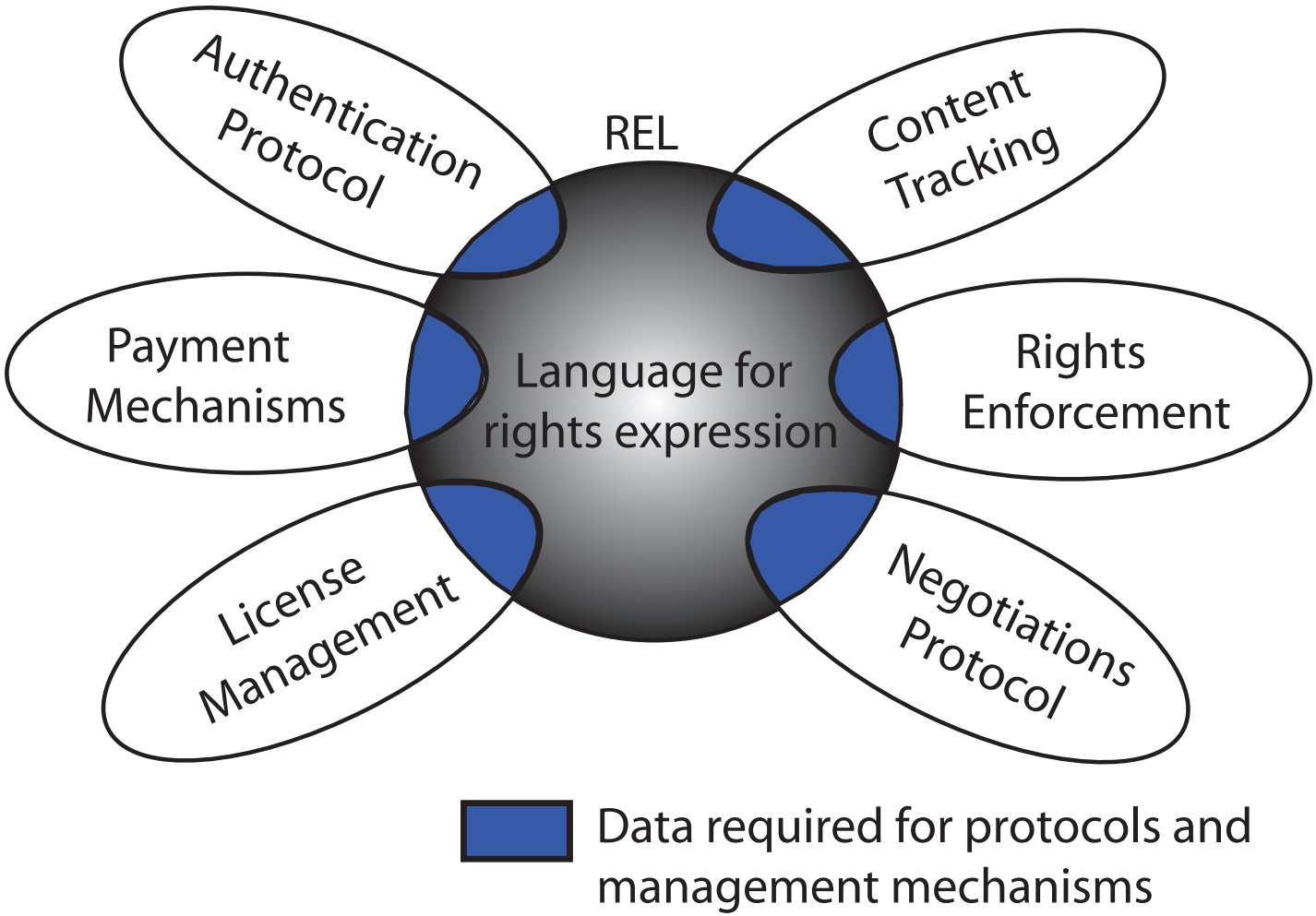
1. **The core REL should only contain the rights model.**
2. **The core REL should be stateless.**
3. **The core REL should be language neutral.**
4. **REL primitives, and DRM services in general, should refer generically to the services they use.**
5. **A DRM service should only know what it absolutely needs to know in order to complete its task.**

Recommendations

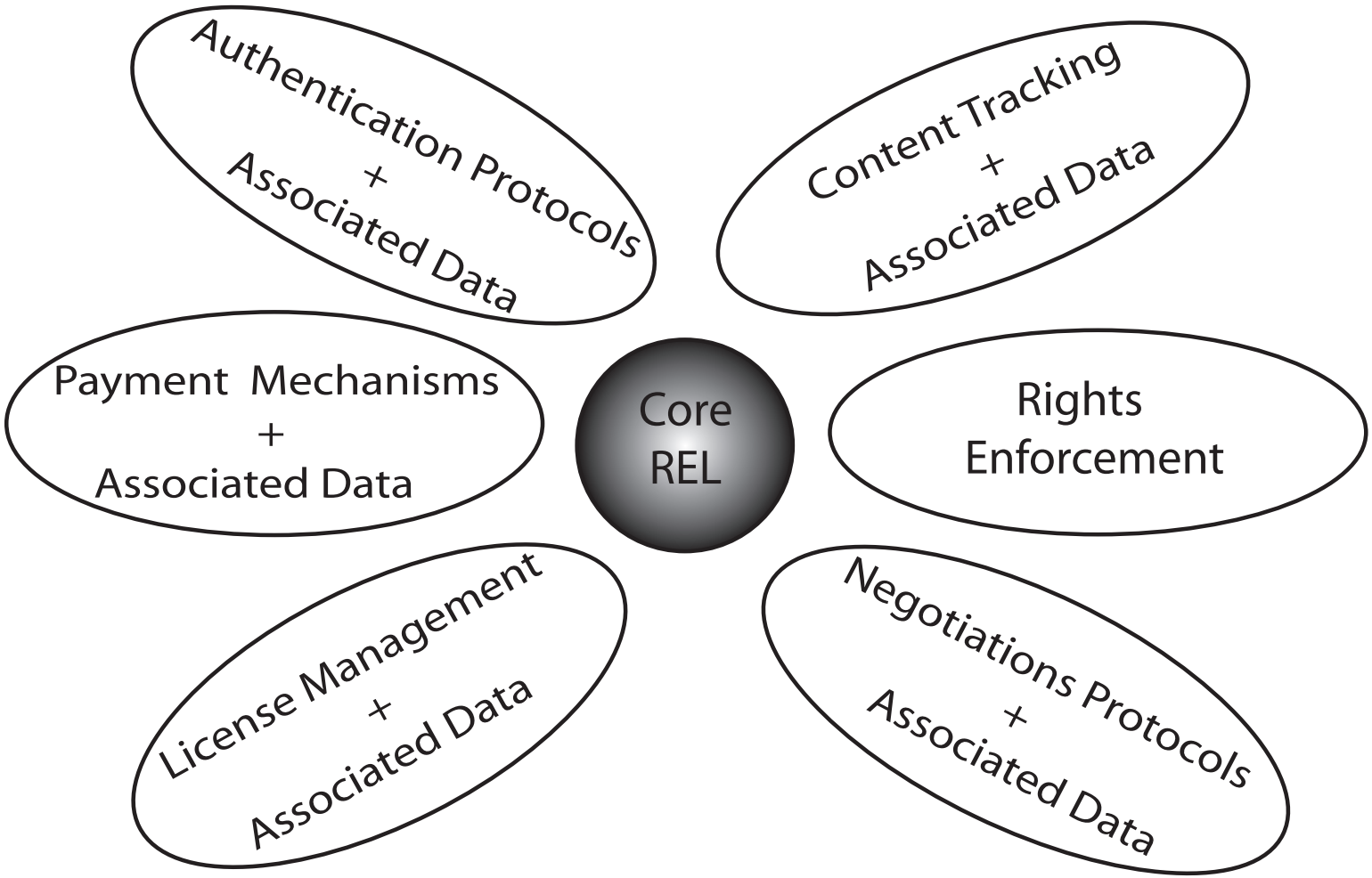
Design principles related to RELs and DRM services that will facilitate the creation of more complicated DRM systems through the composition of middleware services from disparate vendors.

1. **The core REL should only contain the rights model.**
2. **The core REL should be stateless.**
3. **The core REL should be language neutral.**
4. **REL primitives, and DRM services in general, should refer generically to the services they use.**
5. **A DRM service should only know what it absolutely needs to know in order to complete its task.**
6. **A DRM service should encapsulate the operations it performs, along with the data used by these operations.**

DRM Services – Simplified Core REL

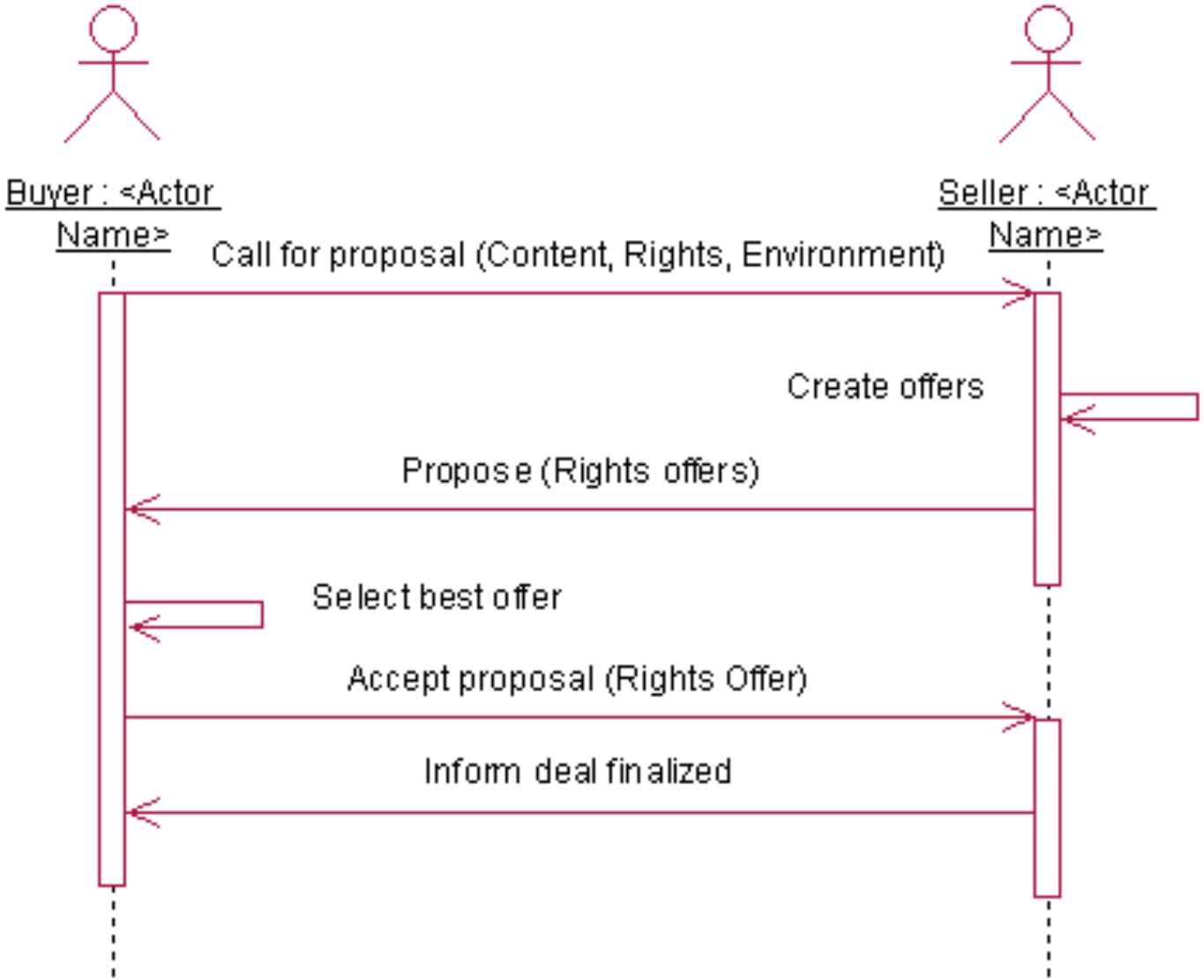


DRM Services – Simplified Core REL



An Example DRM Scenario

Rights negotiations associated with a content purchase:



An Example DRM Scenario

- This is a simplification of the Foundation for Intelligent Agents (FIPA) Contract Net Interaction (CNI) Protocol.

An Example DRM Scenario

- This is a simplification of the Foundation for Intelligent Agents (FIPA) Contract Net Interaction (CNI) Protocol.
- This scenario cannot be entirely implemented in either XrML or ODRL.

An Example DRM Scenario

- This is a simplification of the Foundation for Intelligent Agents (FIPA) Contract Net Interaction (CNI) Protocol.
- This scenario cannot be entirely implemented in either XrML or ODRL.
- We will consider two implementations:
  - Extension-based – include the semantics of the protocol and supporting data representation with the REL, and then develop a negotiations protocol that understands and uses the extended REL.

An Example DRM Scenario

- This is a simplification of the Foundation for Intelligent Agents (FIPA) Contract Net Interaction (CNI) Protocol.
- This scenario cannot be entirely implemented in either XrML or ODRL.
- We will consider two implementations:
  - Extension-based – include the semantics of the protocol and supporting data representation with the REL, and then develop a negotiations protocol that understands and uses the extended REL.
  - Middleware-based – the semantics of the protocol and supporting data are kept separate from the REL, and therefore the negotiations protocol is developed independent of the REL.

Protocol-integrated REL

The Buyer issues a call-for-proposal request:

```
<call-for-proposal>
    <environment> .......... </environment>
    <rights>
        <party> ............ </party>
        <content> .......... </content>
        <permission>.........</permission>
    </rights>
</call-for-proposal>
```

Protocol-integrated REL

The Seller analyze the call and creates a set of offers consistent with its policies:

```
<propose>
     <offer1>
          <rights>.....</rights>
     </offer1>
     <offer2>
          <rights>......</rights>
     </offer2>
</propose>
```

Protocol-integrated REL

The Buyer selects a particular offer by issuing an accept-proposal request informing the Seller which offer will be accepted:

```
<accept-proposal>
    <offer>
        <rights>......</rights>
    </offer>
</accept-proposal>
```

Protocol-integrated REL

- The tags associated with the performatives: `call-for-proposal`, `propose` and `accept-proposal` would need to be defined in the extension.

Protocol-integrated REL

- The tags associated with the performatives: `call-for-proposal`, `propose` and `accept-proposal` would need to be defined in the extension.
- In fact, these are only a subset of the performatives used in the CNI protocol. Thus, other tags would need to be defined.

Protocol-integrated REL

- The tags associated with the performatives: `call-for-proposal`, `propose` and `accept-proposal` would need to be defined in the extension.
- In fact, these are only a subset of the performatives used in the CNI protocol. Thus, other tags would need to be defined.
- CNI is just one of the protocols specified by FIPA. Others, such as the Request Interaction Protocol and the Query Interaction Protocol would also be useful in DRM settings. Tags for these protocols would also need to be defined.

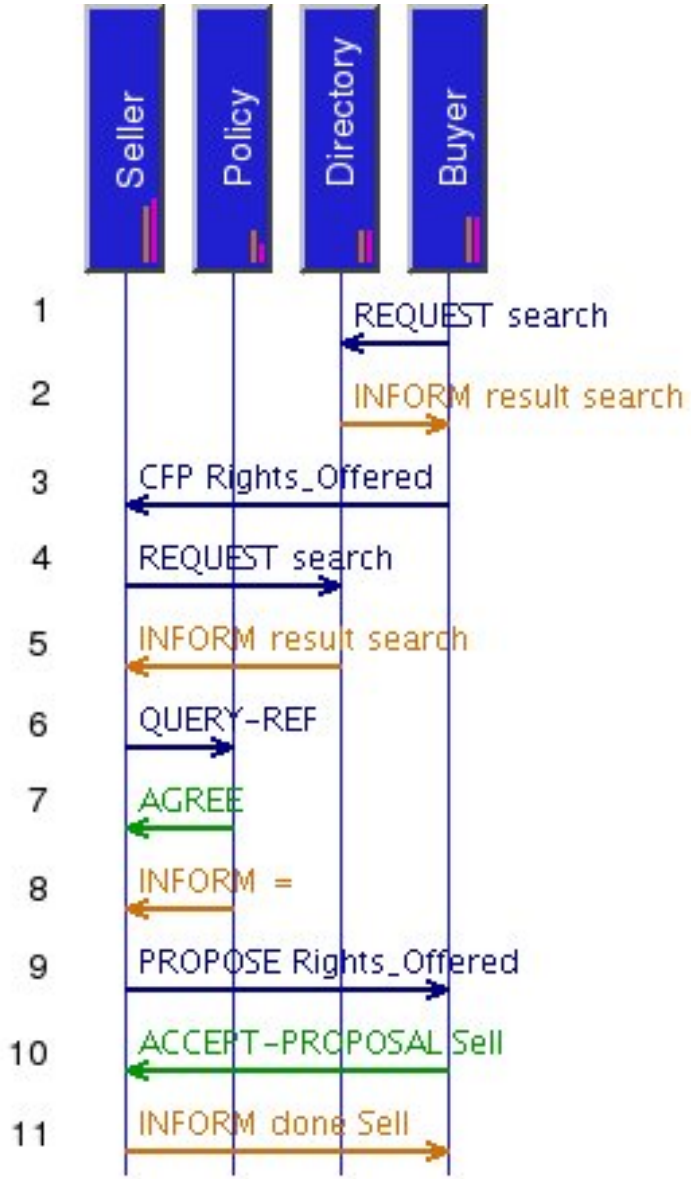
Middleware Approach

- In this approach, the REL is used purely for rights expression, and therefore the semantics of the protocol are kept out of the REL.

Middleware Approach

- In this approach, the REL is used purely for rights expression, and therefore the semantics of the protocol are kept out of the REL.
- We implemented this using an agent-based architecture.
  - Buyer agent
  - Seller agent
  - Policy agent

Middleware Approach



Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.

Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.
- Since the protocol is tied to the REL, the system must be developed in a monolithic manner.

Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.
- Since the protocol is tied to the REL, the system must be developed in a monolithic manner.
- In the second approach, the protocol and the REL were separated, and rights were simply a payload used to carry out the negotiations.

Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.
- Since the protocol is tied to the REL, the system must be developed in a monolithic manner.
- In the second approach, the protocol and the REL were separated, and rights were simply a payload used to carry out the negotiations.
- Only the Policy agent needed to understand the semantics of the rights — the other agents only need to understand the semantics of the protocol.

Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.
- Since the protocol is tied to the REL, the system must be developed in a monolithic manner.
- In the second approach, the protocol and the REL were separated, and rights were simply a payload used to carry out the negotiations.
- Only the Policy agent needed to understand the semantics of the rights — the other agents only need to understand the semantics of the protocol.
- Thus, if the rights model is changed, and only the Policy agent would need to be changed.

Summary

- Integration of the protocol semantics within the REL imposes severe restrictions – any change in the protocol would require changes in the REL.
- Since the protocol is tied to the REL, the system must be developed in a monolithic manner.
- In the second approach, the protocol and the REL were separated, and rights were simply a payload used to carry out the negotiations.
- Only the Policy agent needed to understand the semantics of the rights — the other agents only need to understand the semantics of the protocol.
- Thus, if the rights model is changed, and only the Policy agent would need to be changed.
- This makes is easy to package these DRM negotiations services as middleware.

Conclusions

- Current RELs incorporate far more functionality than pure rights expression.

Conclusions

- Current RELs incorporate far more functionality than pure rights expression.
- We have demonstrated how one set of DRM services could be implemented as middleware using a simplified REL, as well as the benefits obtained by doing so.

Conclusions

- Current RELs incorporate far more functionality than pure rights expression.
- We have demonstrated how one set of DRM services could be implemented as middleware using a simplified REL, as well as the benefits obtained by doing so.
- Other DRM services, e.g., license management, payment services, authentication, security, etc., would similarly benefit from this approach.

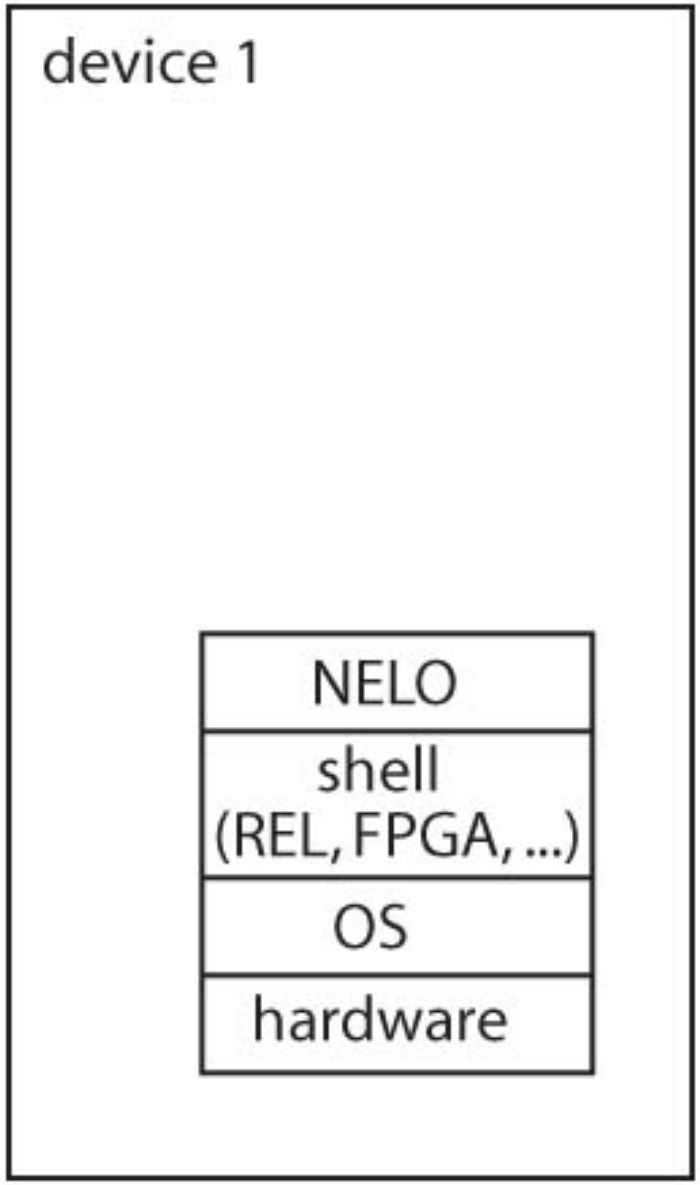
Conclusions

- Current RELs incorporate far more functionality than pure rights expression.
- We have demonstrated how one set of DRM services could be implemented as middleware using a simplified REL, as well as the benefits obtained by doing so.
- Other DRM services, e.g., license management, payment services, authentication, security, etc., would similarly benefit from this approach.
- The key ingredient involves separating rights expression from DRM services.

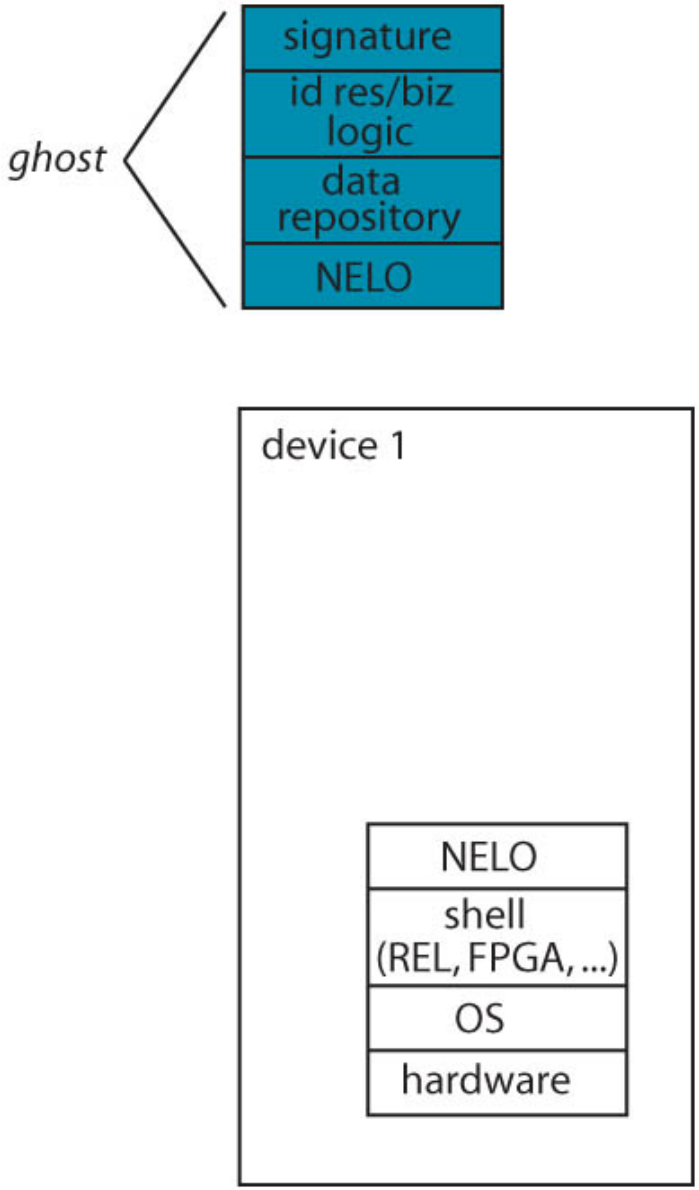
Conclusions

- Current RELs incorporate far more functionality than pure rights expression.
- We have demonstrated how one set of DRM services could be implemented as middleware using a simplified REL, as well as the benefits obtained by doing so.
- Other DRM services, e.g., license management, payment services, authentication, security, etc., would similarly benefit from this approach.
- The key ingredient involves separating rights expression from DRM services.
- The resulting simplified core REL greatly facilitates formal analysis.

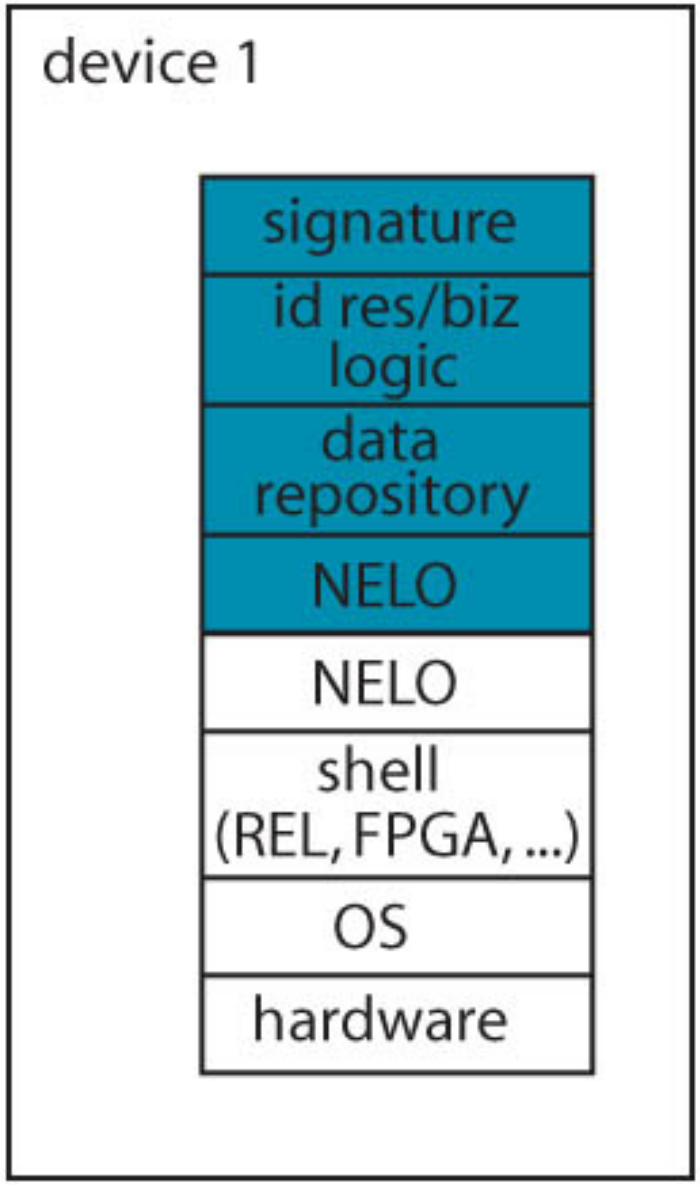
Example



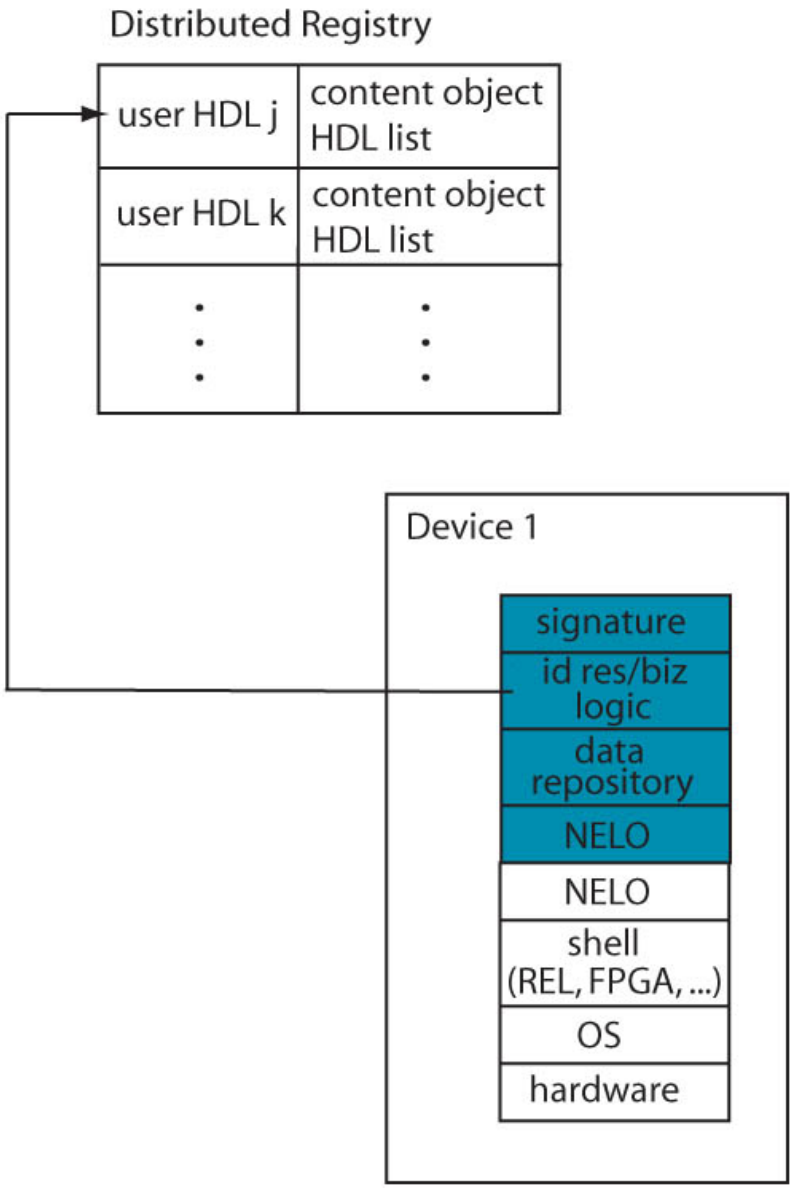
Example


ghost
signature
id res/biz logic
data repository
NELO
device 1
NELO
shell (REL, FPGA, ...)
OS
hardware

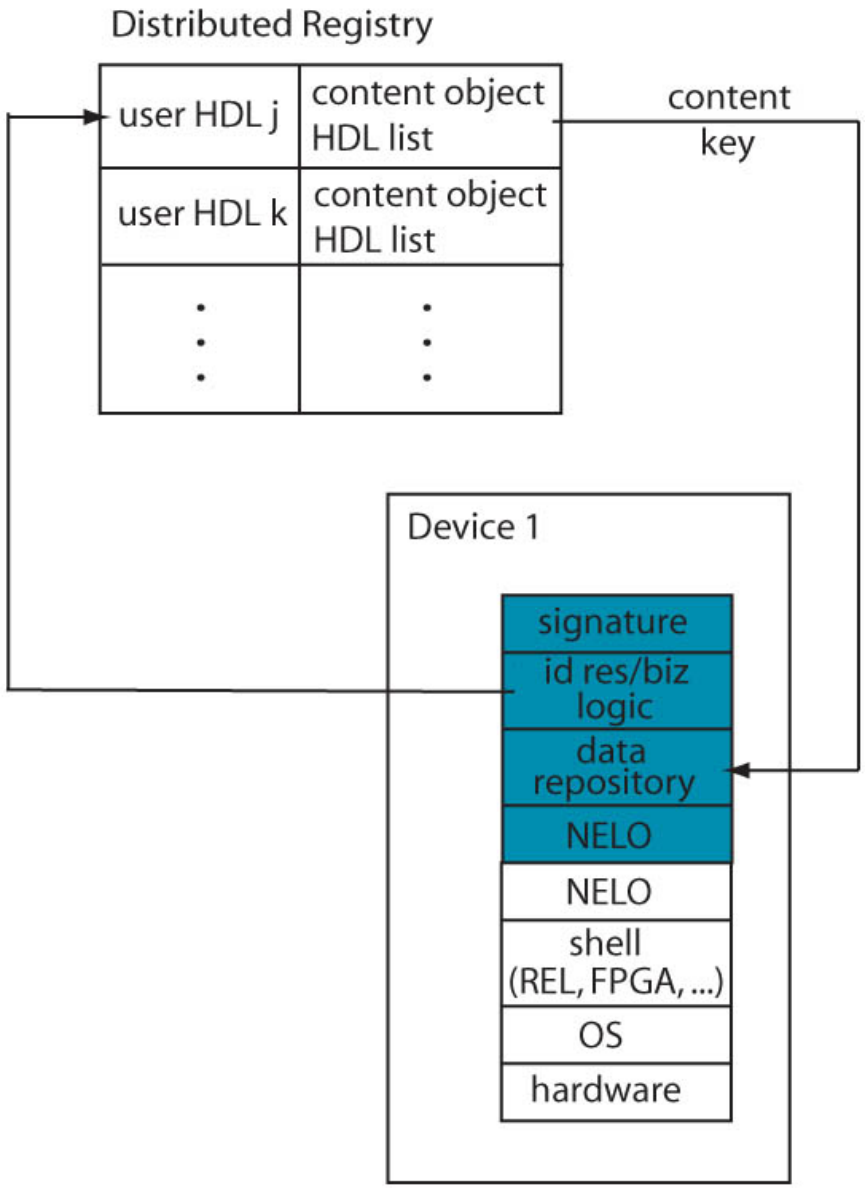
Example


device 1
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware

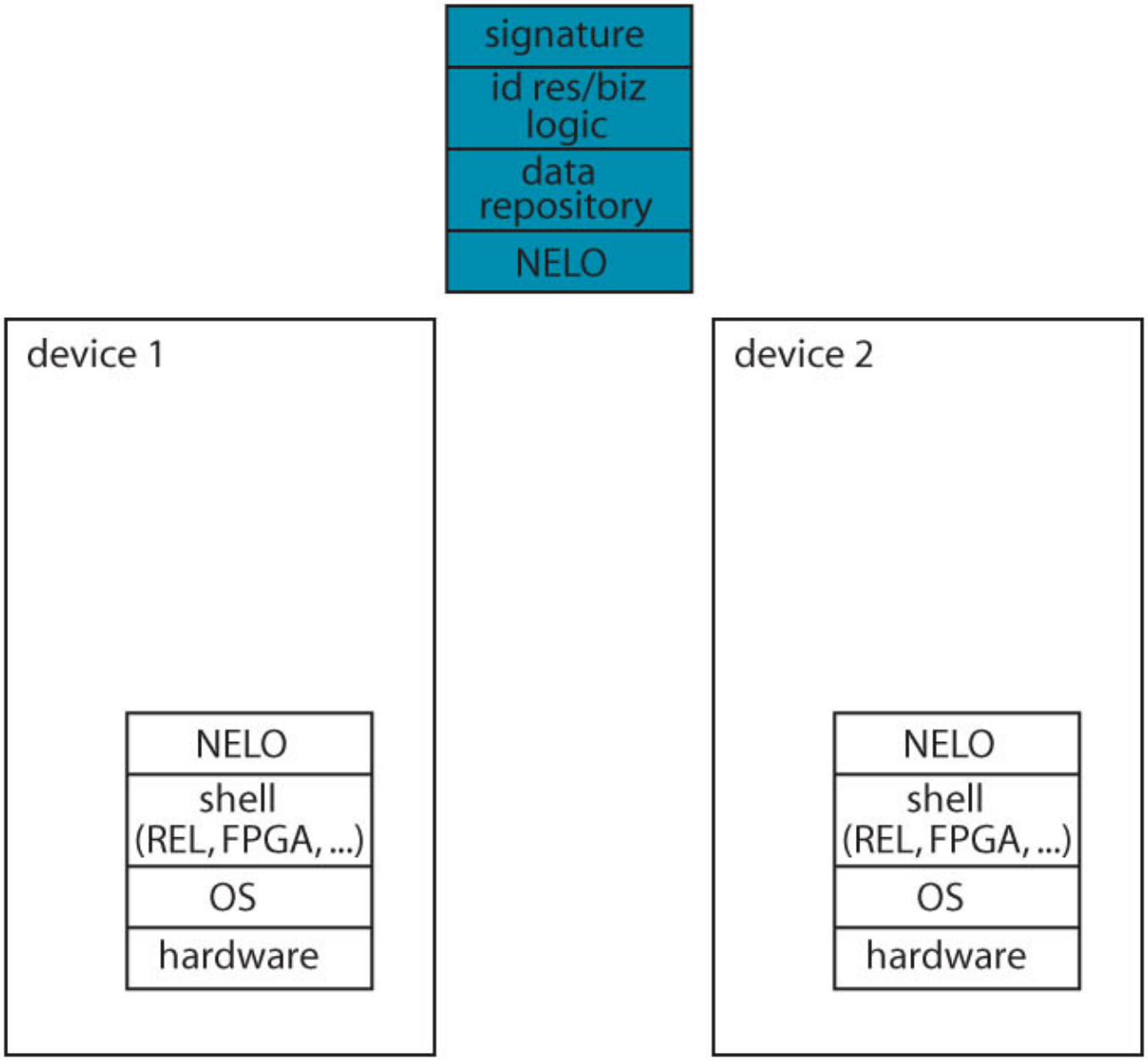
Example


Distributed Registry
user HDL j
content object HDL list
user HDL k
content object HDL list
Device 1
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware

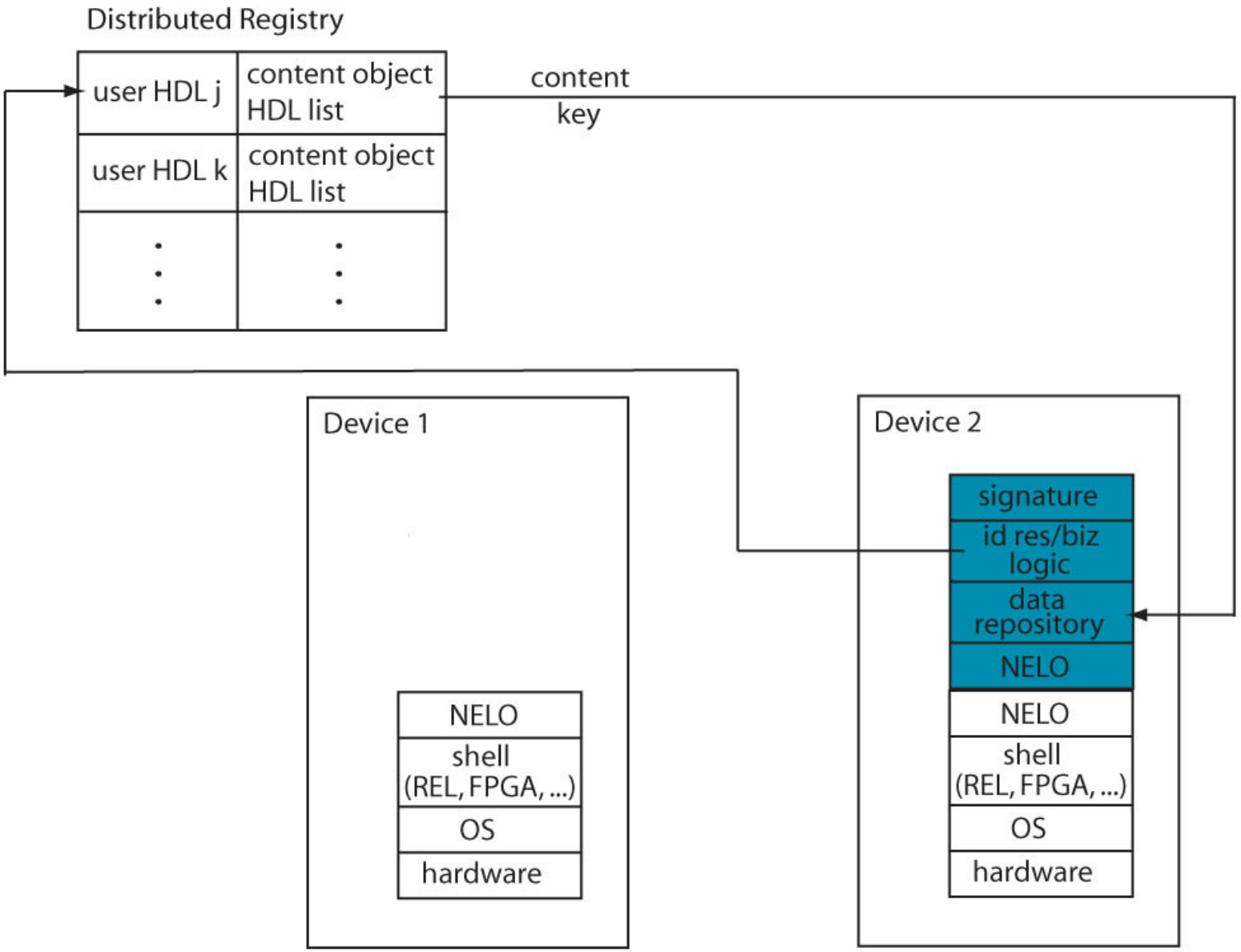
Example


Distributed Registry
user HDL j
content object HDL list
user HDL k
content object HDL list
content key
Device 1
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware

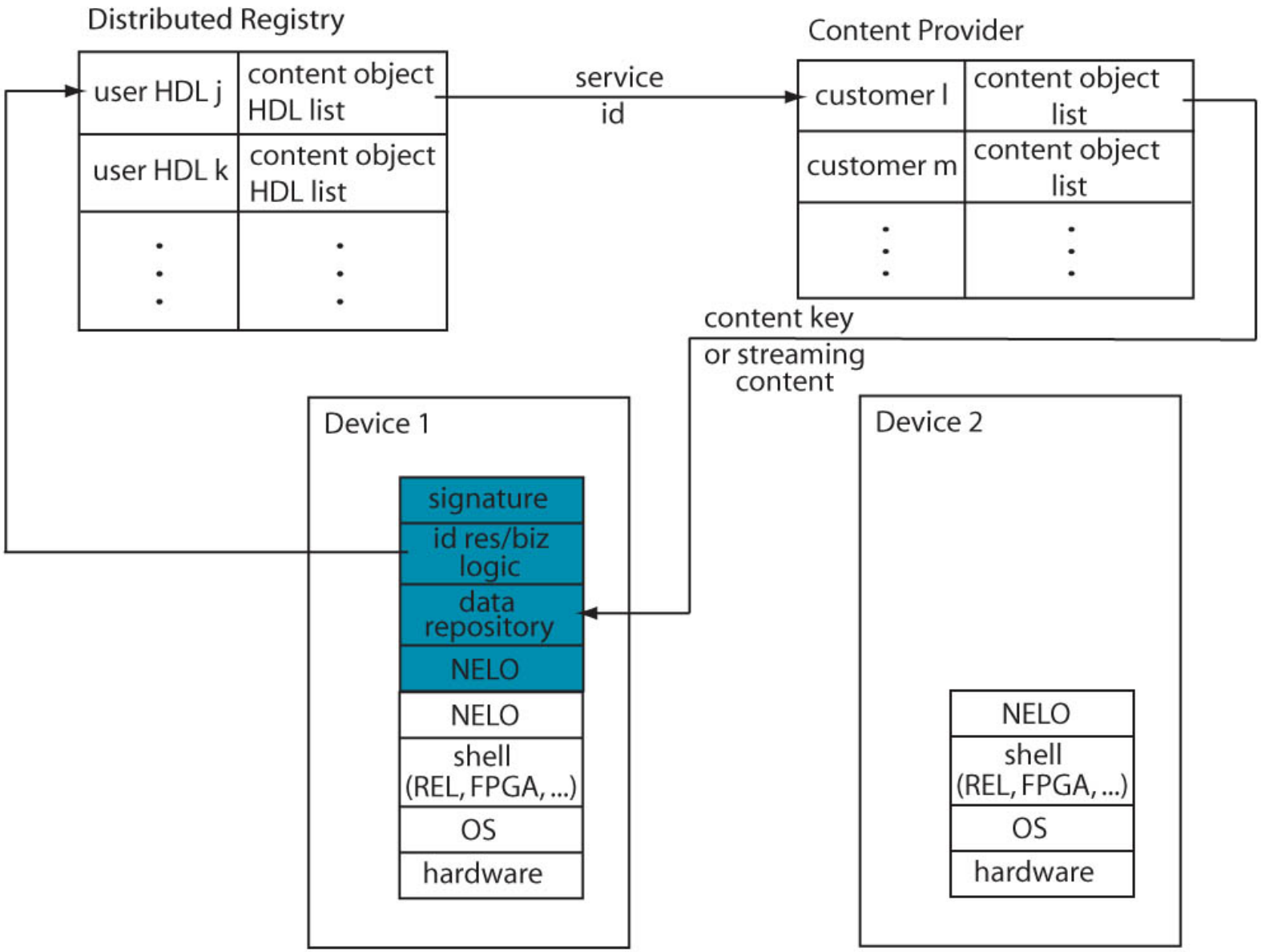
Example


signature
id res/biz logic
data repository
NELO
device 1
NELO
shell (REL, FPGA, ...)
OS
hardware
device 2
NELO
shell (REL, FPGA, ...)
OS
hardware

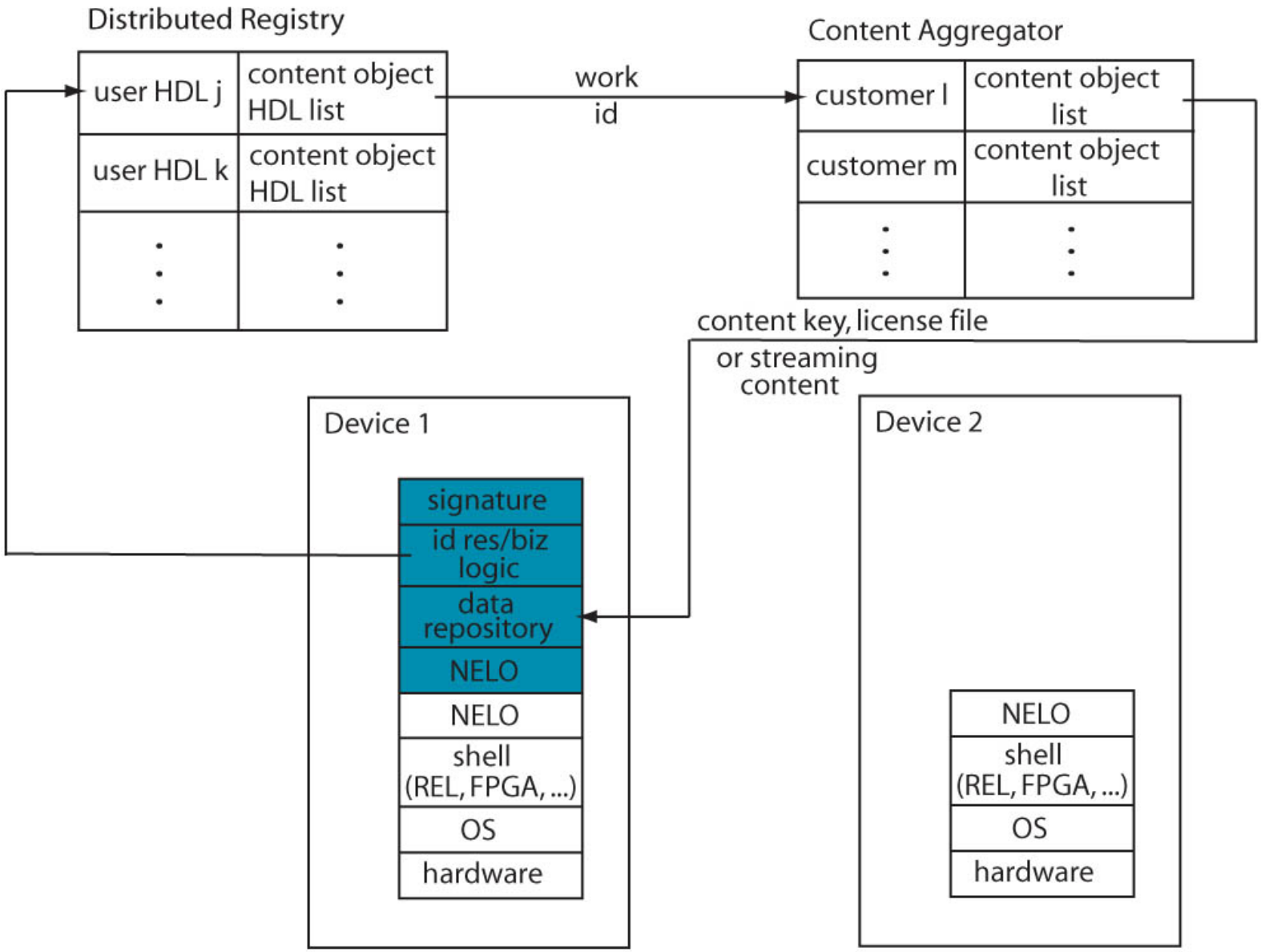
Example


Distributed Registry
user HDL j
content object HDL list
user HDL k
content object HDL list
content key
Device 1
NELO
shell (REL, FPGA, ...)
OS
hardware
Device 2
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware

Example


Distributed Registry
user HDL j
content object HDL list
user HDL k
content object HDL list
service id
Content Provider
customer l
content object list
customer m
content object list
content key or streaming content
Device 1
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware
Device 2
NELO
shell (REL, FPGA, ...)
OS
hardware

Example


Distributed Registry
user HDL j
content object HDL list
user HDL k
content object HDL list
work id
Content Aggregator
customer l
content object list
customer m
content object list
content key, license file
or streaming content
Device 1
signature
id res/biz logic
data repository
NELO
NELO
shell (REL, FPGA, ...)
OS
hardware
Device 2
NELO
shell (REL, FPGA, ...)
OS
hardware