Security in Applications: Multimedia Content (DRM)

2024/25 Q2

Jaime Delgado

DAC - UPC

DMAG DISTRIBUTED MULTIMEDIA APPLICATIONS GROUP

Intellectual rights for multimedia

Intellectual rights for multimedia content:

Intellectual property

Standards for licenses and contracts

DRM and the Web

Intellectual property

- Owned by “creators” (rights holders, *titulares de derechos*).
- Article 27 of the Universal Declaration of Human Rights.
- *“Ley de Propiedad Intelectual”*.
- WIPO (World Intellectual Property Organization).
- Industrial property + Author right (Copyright).
- Moral rights: paternity, integrity, disclosure *(divulgación)*, withdrawal, …
- Economic rights *(patrimoniales)*: reproduction, distribution, public communication, transformation.

  Exclusive, Remunerated
- Related rights *(conexos)*: performers (intérpretes), broadcasters, …

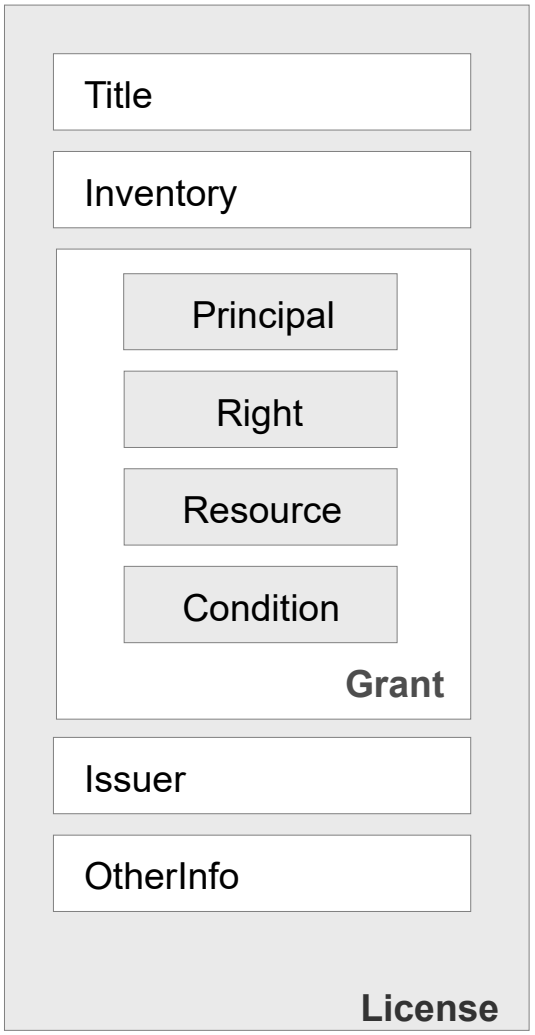
Intellectual rights for multimedia

Intellectual rights for multimedia content:

- Intellectual property
- **Standards for licenses and contracts**
- **DRM and the Web**

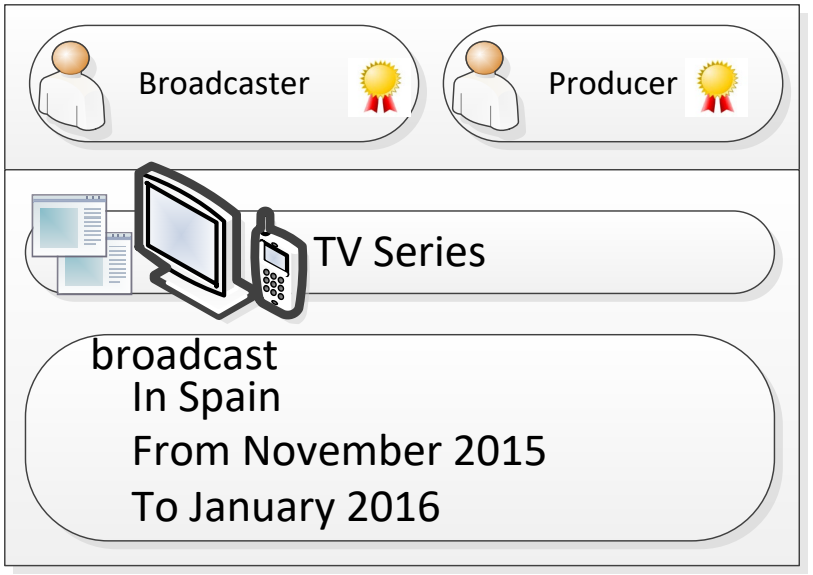
MPEG-21 Part 5: REL

- Rights Expression Language (REL)
  - Language (XML-based) for expressing rights and conditions for using content
  - Most important concept: **License**



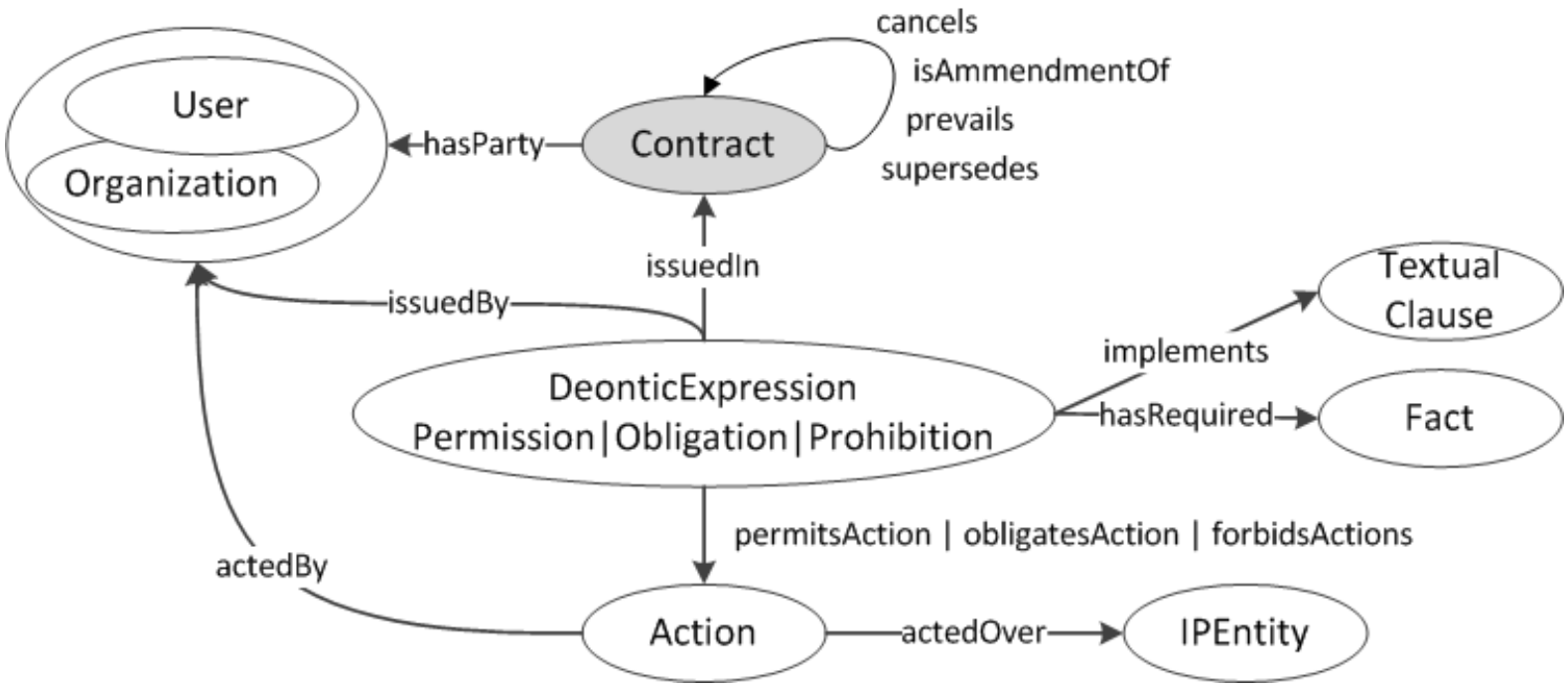
MPEG-21 Part 20: CEL

- Contracts Expression Language (CEL)
  - Language (XML-based) for expressing contracts defining actions and constraints for using content → Extends REL
  - Most important concept: **Contract**



MPEG-21 Part 21: MCO

- Media Contract Ontology (MCO)
  - Ontology model of CEL contracts
  - Deontic expressions: *Permission, Prohibition, Obligation*



Intellectual rights for multimedia

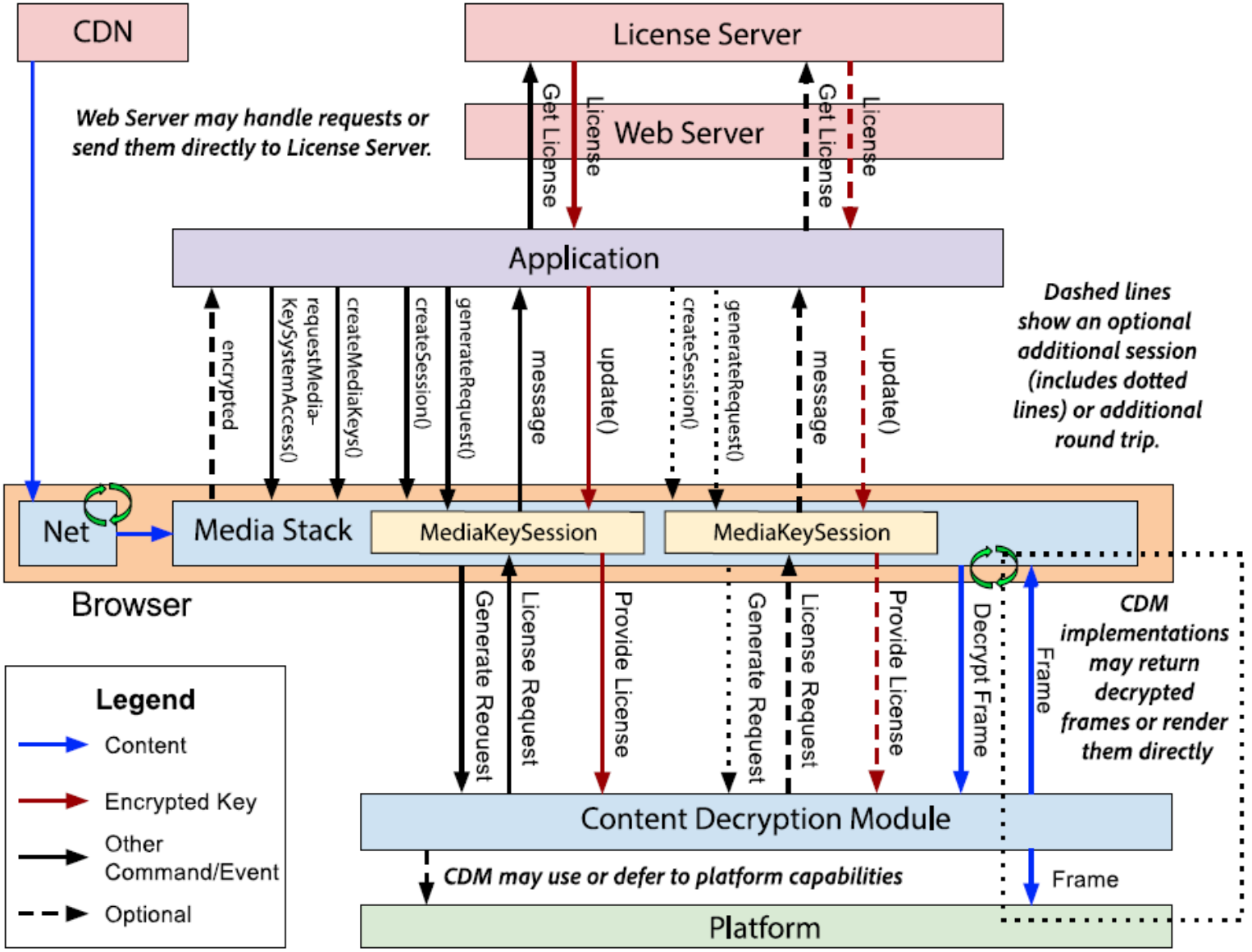
Intellectual rights for multimedia content:

- Intellectual property
- Standards for licenses and contracts
- **DRM and the Web**

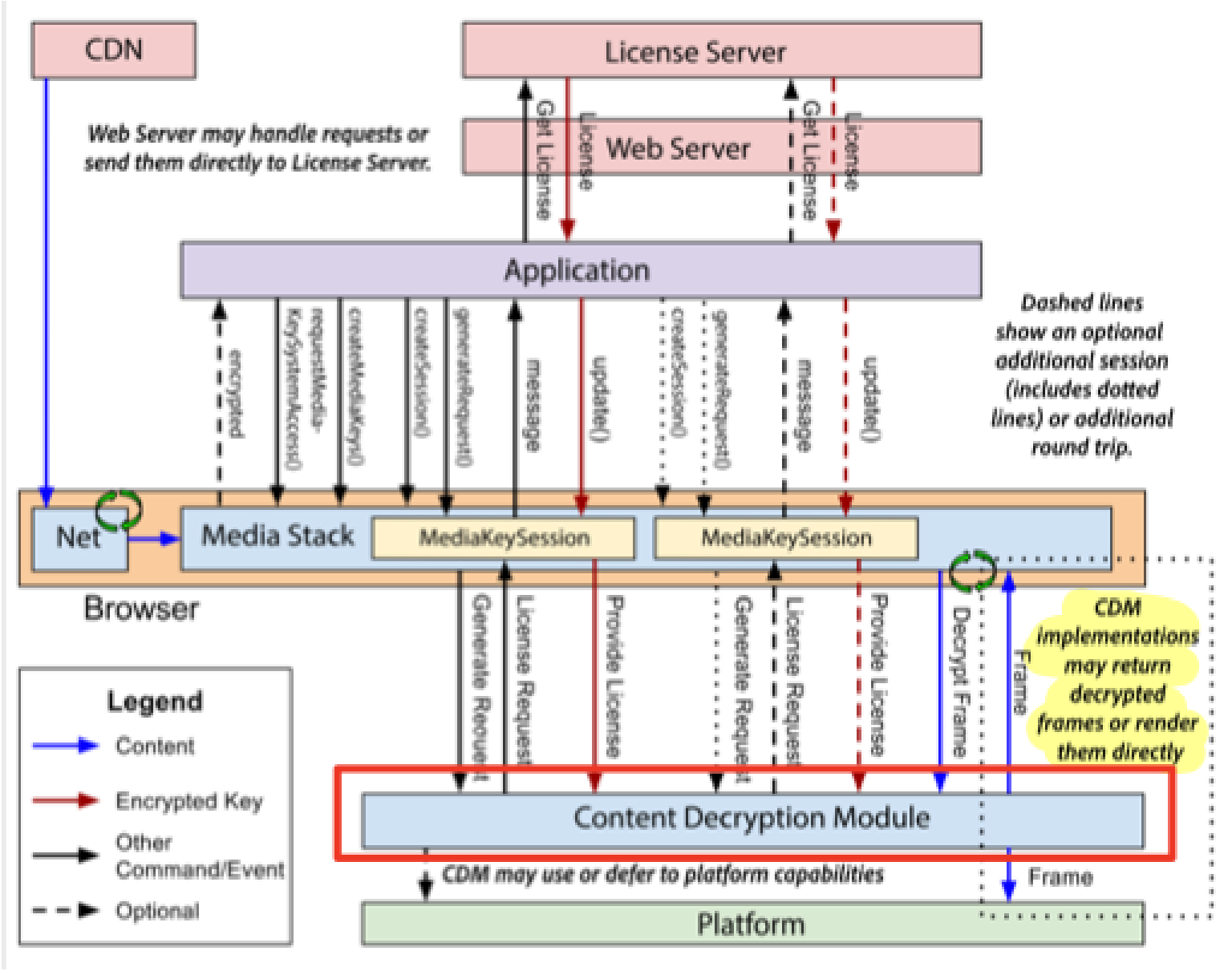
DRM & Web browsers

- **Encrypted Media Extensions (EME)** *(http://www.w3.org/TR/encrypted-media/)*
  - Communication channel between web browsers and DRM (Digital Rights Management) agent software
  - Extends *HTMLMediaElement* providing APIs to control playback of encrypted content
  - License/key exchange controlled by the application
  - Not defining content protection or DRM system
  - The common API supports a simple set of content encryption capabilities

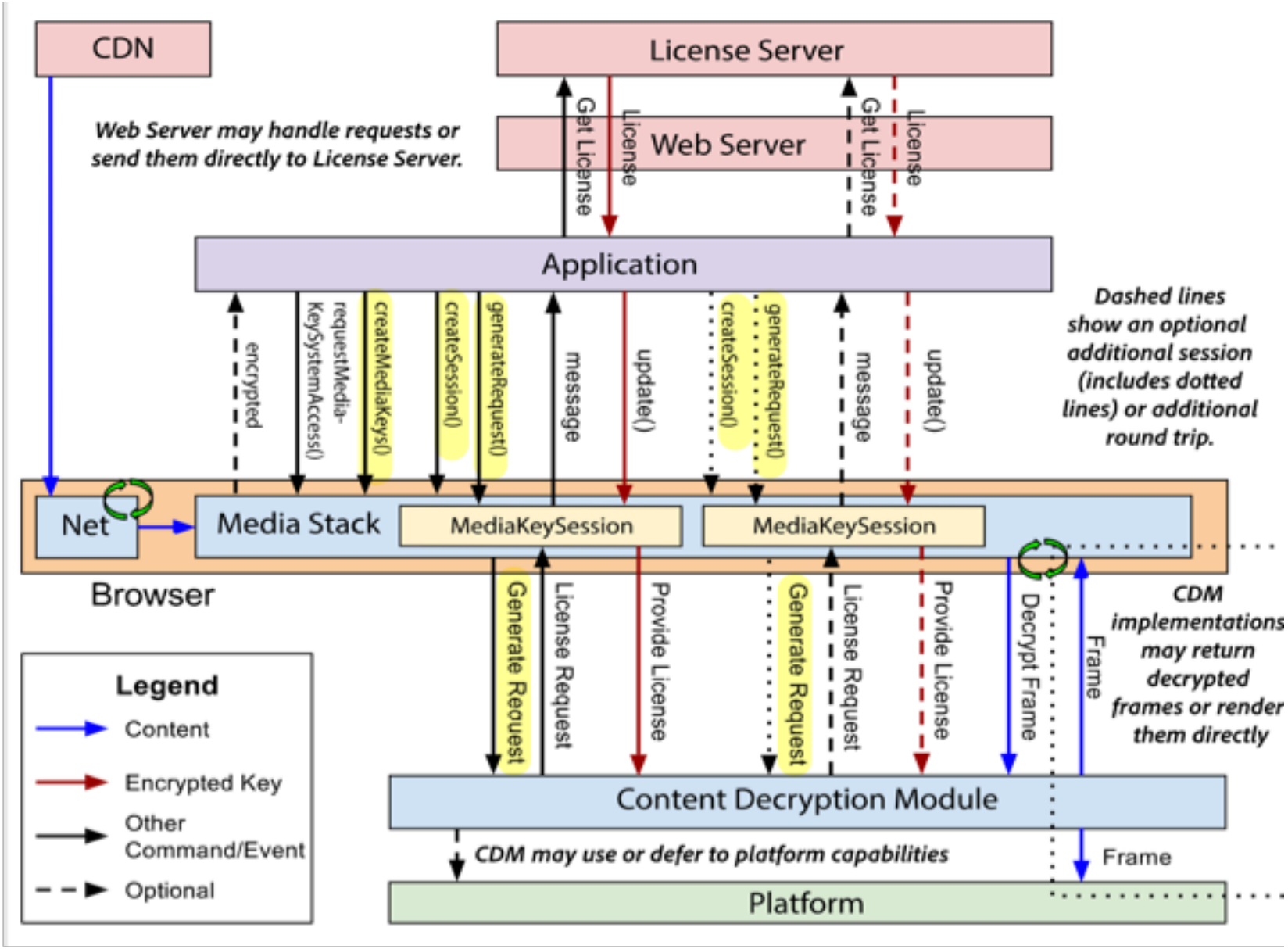
Encrypted Media Extensions (EME)



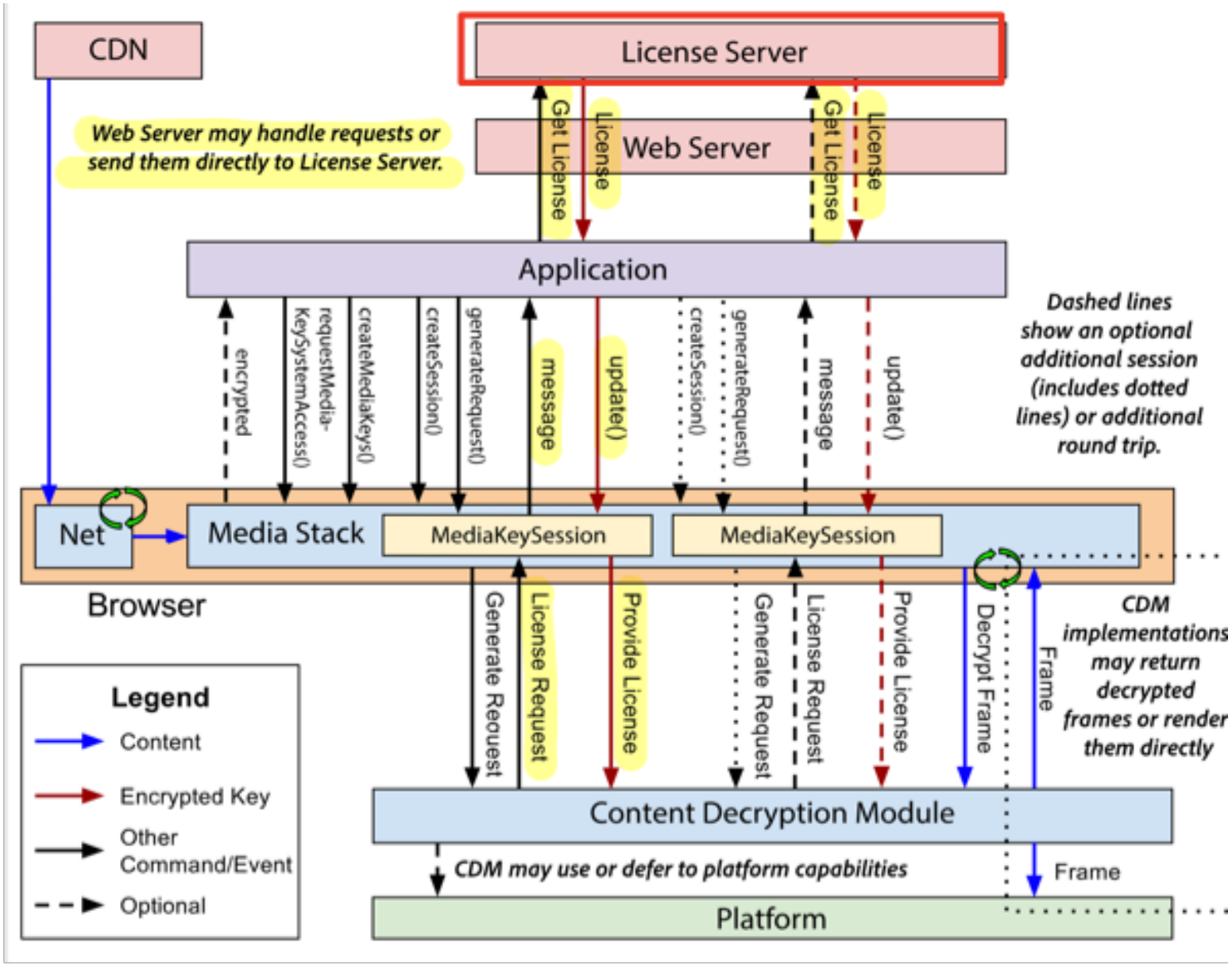
Content Decryption Module



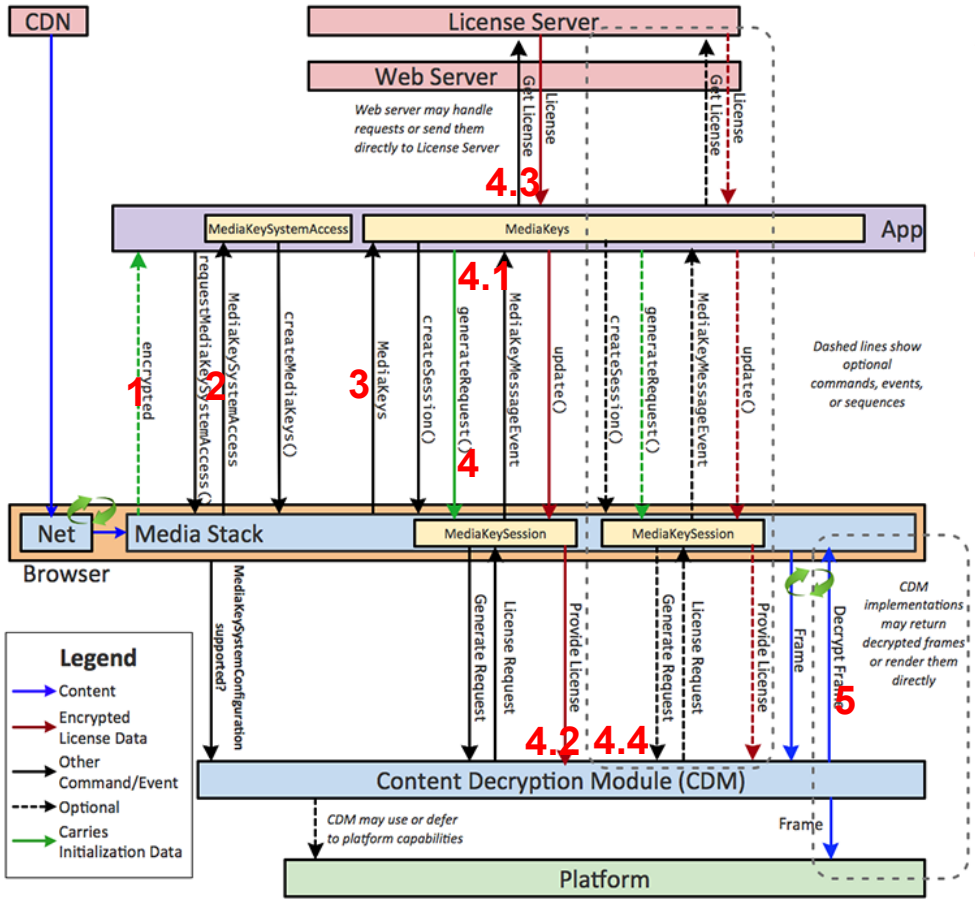
Media Key Sessions



License


CDN
License Server
Web Server may handle requests or send them directly to License Server.
Get License
License
Web Server
Get License
License
Application
encrypted
requestMediaKeySystemAccess()
createMediaKeys()
createSession()
generateRequest()
message
update()
createSession()
generateRequest()
message
update()
Dashed lines show an optional additional session (includes dotted lines) or additional round trip.
Net
Media Stack
MediaKeySession
MediaKeySession
Browser
Generate Request
License Request
Provide License
Generate Request
License Request
Provide License
Decrypt Frame
Frame
CDM implementations may return decrypted frames or render them directly
Legend
Content
Encrypted Key
Other Command/Event
Optional
Content Decryption Module
CDM may use or defer to platform capabilities
Frame
Platform

EME Workflow



1 - Browser notifies the app that encountered encrypted media samples for which it has no appropriate decryption key

2 - App requests access to a DRM system available in the browser

3 - App assigns selected DRM system to an HTMLMediaElement

4 - App creates one or more *key sessions* associated with the selected DRM system

4.1 - App instructs the key session to generate a license request message by providing it with *initialization data*

4.2 - CDM generates a data blob (license request) and delivers it to the app

4.3 - App sends the license request to a license server.

4.4 - Upon receiving a response to its license request, app passes the response message back to the CDM. The CDM adds to the key session any decryption keys contained within the response

5 - CDM and/or browser use keys stored in the key session to decrypt media samples as they are encountered

DRM & Web browsers (2/2)

- **Media Source Extensions (MSE)**
  (https://www.w3.org/TR/media-source/)
  - Extends *HTMLMediaElement* to allow *JavaScript* to generate media streams for playback, independently of how the media is fetched (splicing and buffering model)
  - Defines a *MediaSource* object (with one or more *SourceBuffer* objects, where applications append data segments)
  - *SourceBuffer* objects managed as track buffers for audio, video and text data that is decoded and played
  - No support needed for any particular media format or codec
  - Enable interoperability between user agents and web applications when processing media data