



**INTERNATIONAL AGE RATING COALITION**

---

# IARC Rating System

## Store Integration Specifications

## 1 Table of Contents

1	Table of Contents .....	2
2	Introduction .....	4
2.1	Overview.....	4
2.2	Vocabulary.....	4
3	Configuration Options.....	6
3.1	Core IARC Functions .....	6
3.2	IARC Platforms.....	6
3.3	IARC Hosted Configuration Overview.....	7
3.3.1	Questionnaire and Rating Logic Process .....	7
3.3.2	Components Used .....	7
3.4	Store Hosted Configuration Overview.....	7
3.4.1	Questionnaire and Rating Logic Process .....	7
3.4.2	Components Used .....	7
3.5	Configurations Workflow Comparison .....	8
3.6	Ratings Portability Workflow.....	9
3.7	Product Rating Flow: IARC Hosted Configuration .....	10
3.8	Product Rating Flow: Store Hosted Configuration .....	11
3.9	Background Processing Web Services .....	12
4	IARC Website.....	12
4.1	Overview.....	12
4.1.1	Store Provided Web Service: PutCert.....	13
4.1.2	Developer Redirect.....	13
4.1.3	Questionnaire Page .....	13
4.1.4	Help Content.....	13
4.1.5	Ratings Preview Page.....	13
4.1.6	Ratings Confirmation Page .....	13
4.1.7	Email to Developer .....	13
4.1.8	Localization.....	14
4.2	Developer Redirect.....	14
4.3	PutCert Web Service.....	15
4.3.1	Workflow .....	15
4.3.2	PutCert Request.....	15
4.3.3	PutCert Response .....	17

5	IARC Web Services.....	19
5.1	Shared Behavior .....	19
5.1.1	Overview.....	19
5.1.2	Security.....	19
5.1.3	Localization.....	19
5.1.4	Date Format and Precision .....	20
5.2	LoadQuestionnaire .....	20
5.2.1	Business Logic.....	20
5.2.2	Localization.....	21
5.2.3	JSON Example: Child Questions.....	21
5.2.4	JSON Example: Help Content.....	25
5.2.5	Call Frequency .....	28
5.2.6	LoadQuestionnaire Request .....	28
5.2.7	LoadQuestionnaire Response.....	28
5.3	SearchCerts.....	32
5.3.1	Business Logic.....	32
5.3.2	SearchCerts Request.....	32
5.3.3	SearchCerts Response .....	32
5.4	AttachToCert .....	34
5.4.1	Business Logic.....	35
5.4.2	AttachToCert Request .....	35
5.4.3	AttachToCert Response .....	37
5.5	CreateCerts.....	37
5.5.1	Business Logic.....	37
5.5.2	CreateCerts Request.....	37
5.5.3	CreateCerts Response .....	41
5.6	GetRatingChanges .....	41
5.6.1	Business Logic.....	42
5.6.2	GetRatingChanges Request .....	42
5.6.3	GetRatingChanges Response.....	43
5.7	UpdateCerts.....	46
5.7.1	Business Logic.....	46
5.7.2	UpdateCerts Request.....	46
5.7.3	UpdateCerts Response .....	47
6	Offline Ratings Calculator.....	48

- 6.1 Overview..... 48
- 6.2 ORC Input..... 48
- 6.3 ORC Output..... 48
- 6.4 Versioning..... 51
- 6.5 Handling Errors & exceptions ..... 51
- 7 Reference Data..... 51
  - 7.1 Message Table ..... 51
  - 7.2 RatingBoard Table ..... 51
  - 7.3 AgeRating Table..... 52
  - 7.4 Descriptor Table ..... 52
  - 7.5 InteractiveElement table ..... 52
- 8 Guide For Display Of Rating Information On Digital Storefronts\* ..... 53
  - 8.1 Optimal Rating Display ..... 53
  - 8.2 Specifications..... 53
  - 8.3 Alternate Rating Display ..... 54

## 2 Introduction

### 2.1 Overview

This document is intended for distribution to digital game/app marketplaces that are considering integrating the International Age Rating Coalition (IARC) rating tool into their developer on-boarding portal. The document will start by providing a brief walk-through of the system and its integration; then drill down into more detailed technical specifications.

IARC has been developed to provide game/app developers a streamlined system to obtain age ratings that consumers recognize, understand and trust. By completing a single multiple-choice questionnaire, developers immediately receive a rating from numerous official regional rating authorities that reflects local cultural norms and ratings standards. Additionally, a generic rating is assigned to all regions that are not represented by a participating rating authority.

IARC ratings are designed to provide concise and consistent information about the content, interactive elements and age appropriateness of games/apps, so that consumers can make informed decisions as to which ones are most suitable. IARC ratings information can also be mapped to device settings or filters that users (particularly parents) can utilize to control the products that may be accessed or downloaded. Participating marketplaces display any or all of the components of a product’s rating prior to download and may utilize them as filtering criteria.

### 2.2 Vocabulary

#### Product

---

This refers to a game or app that is submitted to the IARC tool for age rating.

## Developer

---

The individual or company that develops the product. The IARC system makes no distinction between the publisher and the developer.

## Store

---

The digital marketplace that makes products available to consumers. In this documentation, “store” refers to the partner company itself, regardless of how many separate storefronts this partner makes available to consumers (e.g. on consoles, mobile, websites, etc.).

## IARC

---

IARC stands for International Age Rating Coalition, but within this documentation we also use it to refer to the IARC digital rating system itself.

## Rating Authority

---

An accredited regional participating authority, such as:

- ESRB: The Entertainment Software Rating Board (The Americas)
- PEGI: Pan European Game Information (Europe)
- ClassInd: Classificação Indicativa (Brazil)
- USK: Unterhaltungssoftware Selbstkontrolle (Germany)
- ACB: Australian Classification Board

## Submission

---

The set of all data required in order to generate a rating certificate, e.g. product name, responses to the questionnaire, etc.

## Certificate

---

After submission data is analyzed, ratings are assigned and an official document is generated by IARC containing all rating information for that product.

A certificate can be used at multiple stores for the same product, i.e. a developer does not need to fill out the IARC questionnaire at every store for the same product. This feature is referred to as ratings portability.

## Rating

---

An IARC rating is comprised of up to three parts:

1. The age rating category, which suggests age appropriateness.
2. Content descriptors, which indicate elements that may have triggered a particular age rating and/or may be of interest or concern.
3. Interactive elements, which advise consumers up front that a product includes one or more of the following:
  - exposure to user-generated content

- displaying a user’s location to other users
- allowing in-product digital purchases
- unrestricted internet access
- collection and sharing of personal information with third parties

For each rating authority, one age rating is provided, and zero or more content descriptors are included. Separately, zero or more interactive elements are provided. Every rating authority has its own set of age ratings, and its own set of content descriptors, but interactive elements are global, they don’t belong to any rating authority.

This complete set of IARC rating components will herein be collectively referred to as simply a “rating”. IARC will generate a rating for each participating rating authority, as well as a generic rating that can be used in territories not represented by a participating rating authority.

### 3 Configuration Options

#### 3.1 Core IARC Functions

The two core functions of IARC are the **Questionnaire** and the **Rating Logic**.

- The questionnaire is a set of questions and responses pertaining to the types of content or functionality found within a product. Additionally it includes HTML help content which the user can choose to review.
- The rating logic is applied to the questionnaire responses to generate the ratings.

There are two configurations available for hosting these functions:

- 1) The IARC website hosts the questionnaire, runs the rating logic, and generates the rating certificate.  
*or*
- 2) The store hosts the questionnaire, runs the rating logic to generate the ratings, and calls an IARC web service to pass the questionnaire responses and generated ratings to IARC, which then creates the official rating certificate.

#### 3.2 IARC Platforms

IARC provides three platforms for providing the Questionnaire and Rating Logic functions:

- **IARC Website**
- **IARC Web Services**
- **Offline Ratings Calculator (ORC)** (hosted by the store)

Here is how those platforms are used by the two configuration options:

	IARC Hosted	Store Hosted
Questionnaire hosting	<b>IARC Website</b> hosts the questionnaire	Store hosts the questionnaire
Rating Logic hosting	Rating logic executes on <b>IARC Website</b>	Rating logic executes at the store, using <b>Offline Ratings Calculator</b>
IARC-provided web services	Store calls <b>IARC Web Services</b>	Store calls <b>IARC Web Services</b>
Store-provided web services	Store provides one web service	Store provides no web services

### 3.3 IARC Hosted Configuration Overview

#### 3.3.1 Questionnaire and Rating Logic Process

1. The store sends the developer to the **IARC Website** to complete the **Questionnaire** (in a new browser window).
2. **IARC Website** executes the **Rating Logic** to generate the ratings.
3. When the developer has completed the submission process, IARC calls a web service provided by the store.

#### 3.3.2 Components Used

- IARC Website
- IARC Web Services:
  - SearchCerts
  - AttachToCert
  - UpdateCerts
  - GetRatingChanges
- Store-provided PutCert web service

### 3.4 Store Hosted Configuration Overview

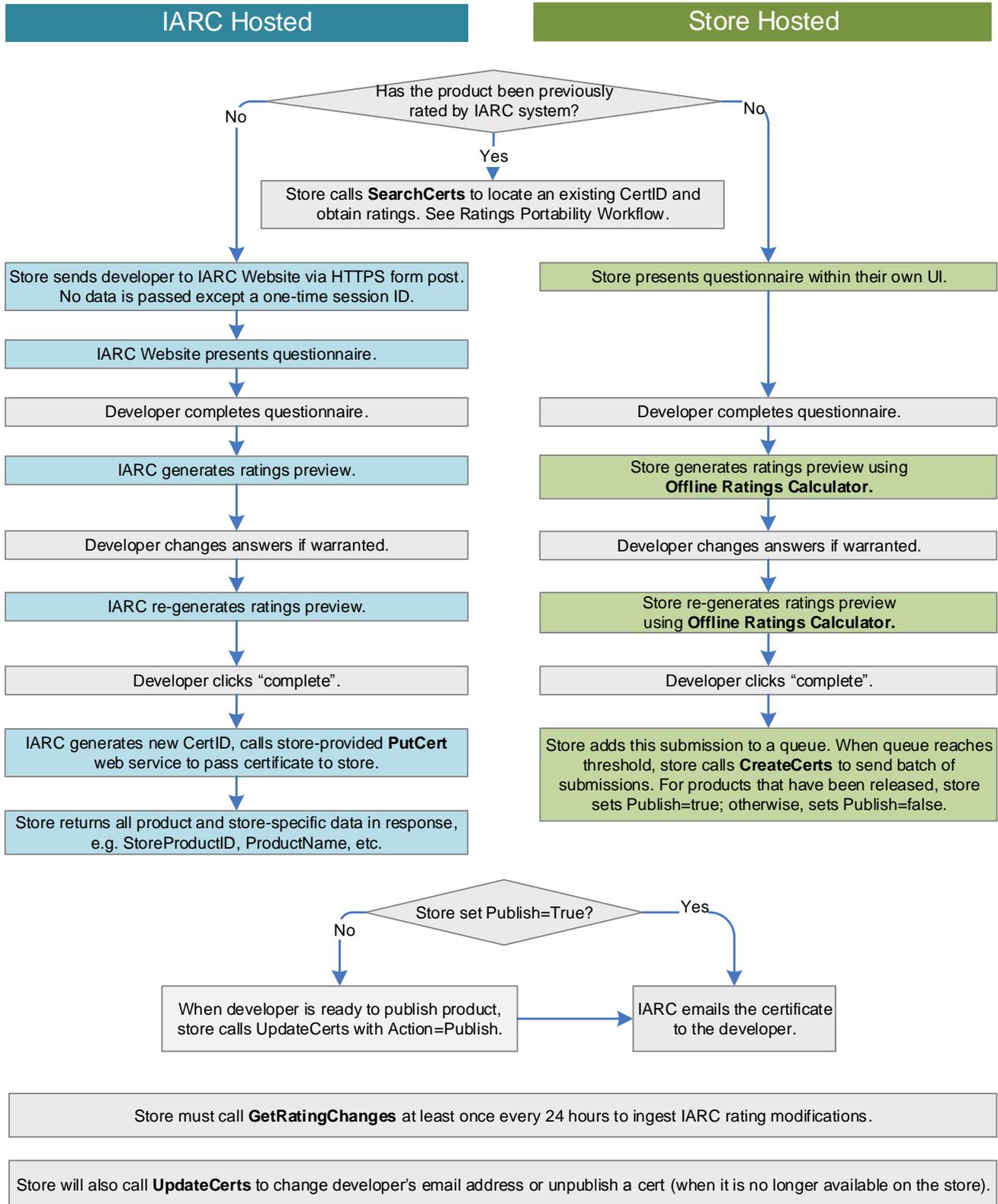
#### 3.4.1 Questionnaire and Rating Logic Process

1. The store calls the LoadQuestionnaire **IARC Web Service** to ingest all questionnaire data (questions, responses, structure, help content, etc.), which is then presented to the developer via the store's own UI.
2. The store uses the **Offline Ratings Calculator** to generate ratings locally.
3. The store maintains a queue of these "offline" submissions, submitting them in batches to the CreateCerts **IARC Web Service**.

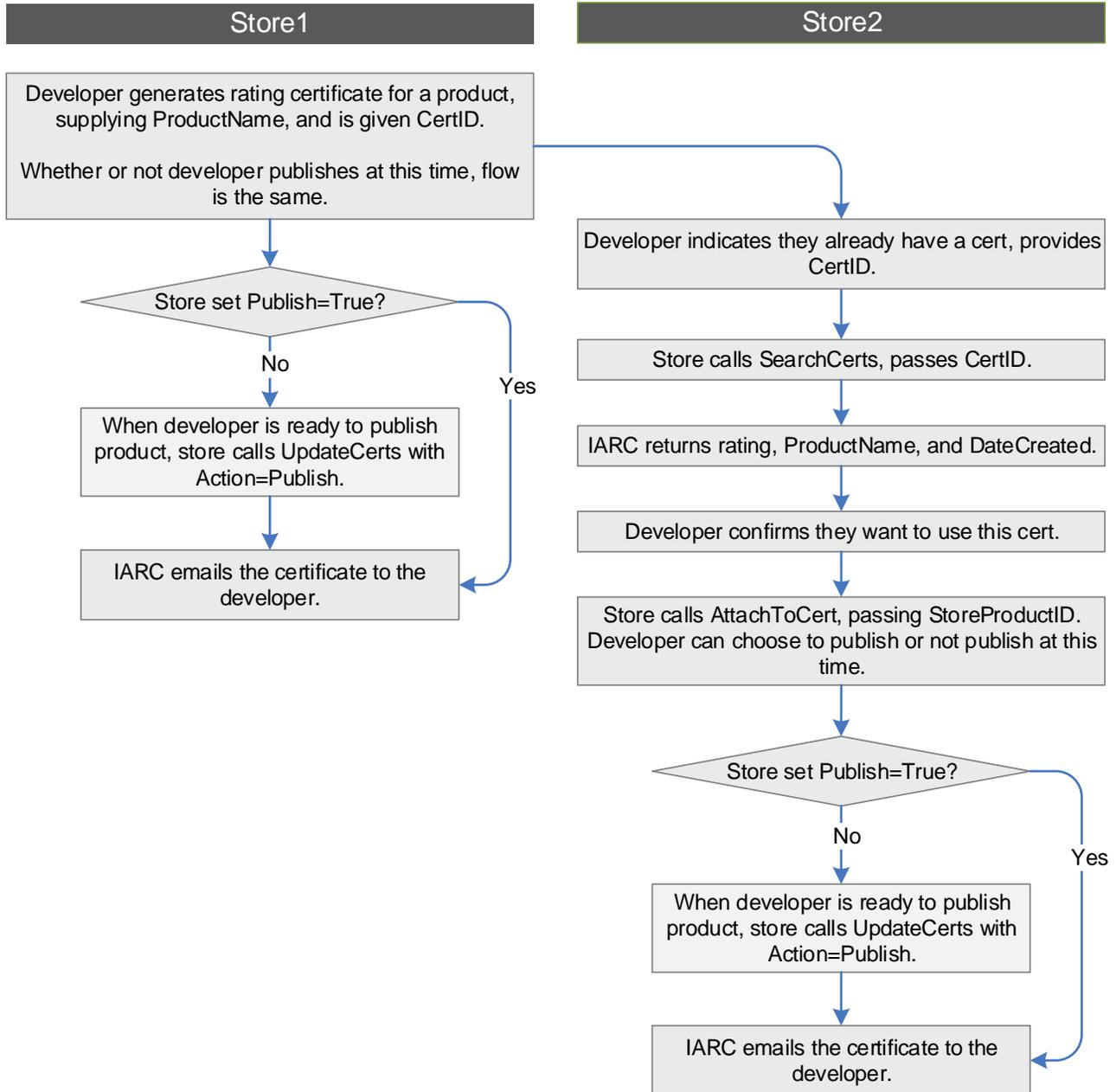
#### 3.4.2 Components Used

- Offline Ratings Calculator
- IARC Web Services (same set as above, with addition of LoadQuestionnaire and CreateCerts):
  - LoadQuestionnaire
  - CreateCerts
  - SearchCerts
  - AttachToCert
  - UpdateCerts
  - GetRatingChanges

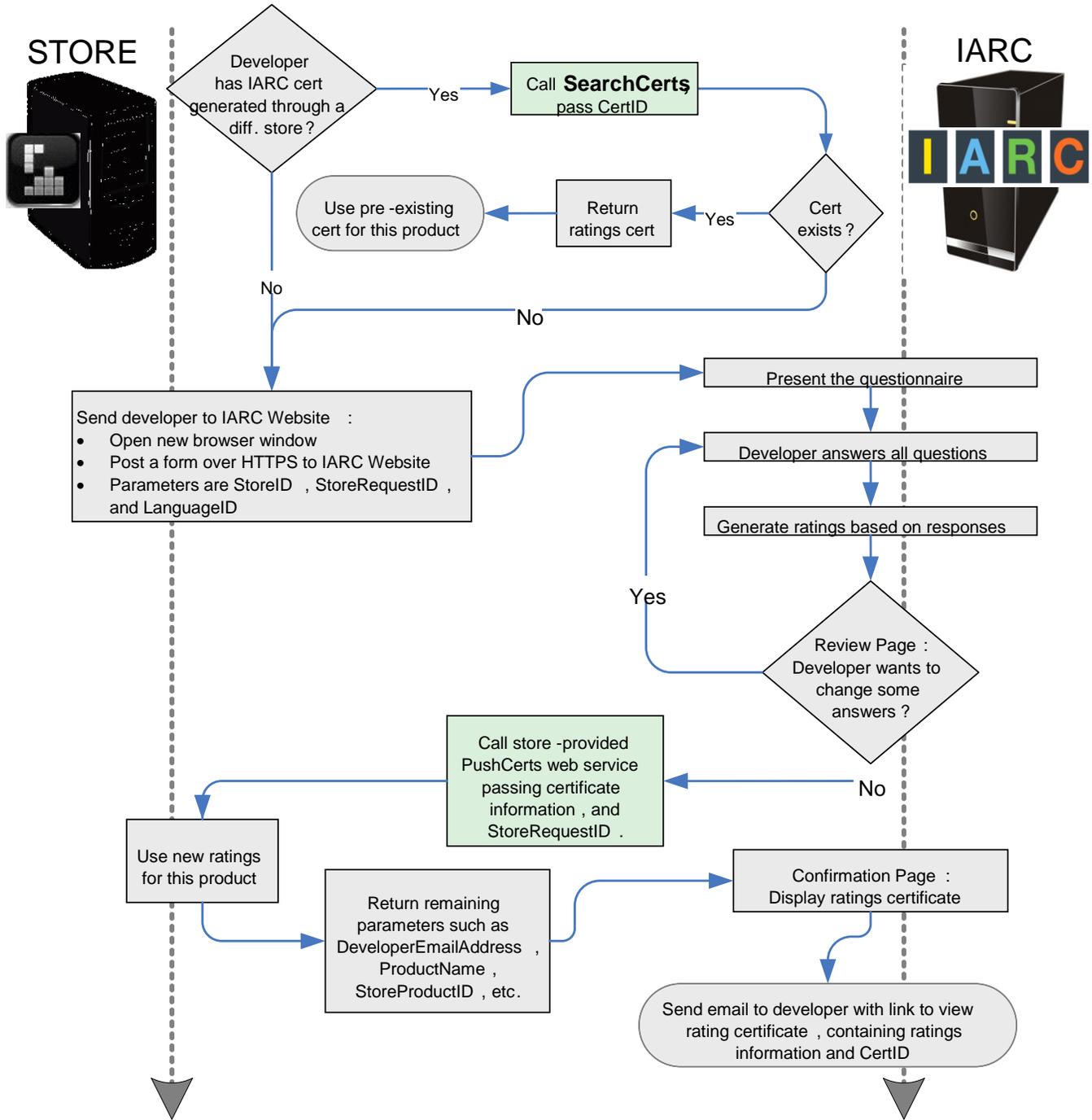
### 3.5 Configurations Workflow Comparison



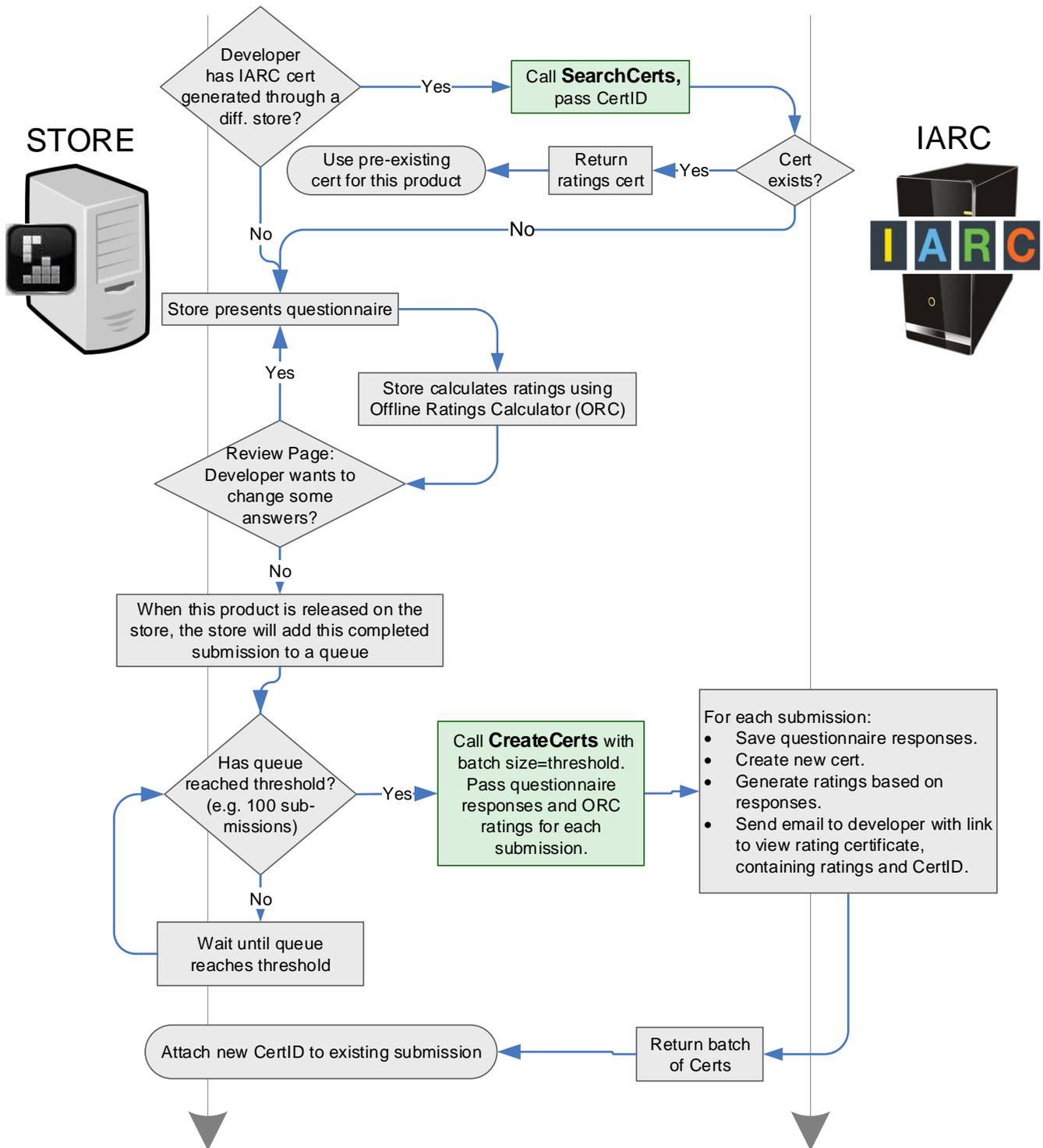
### 3.6 Ratings Portability Workflow



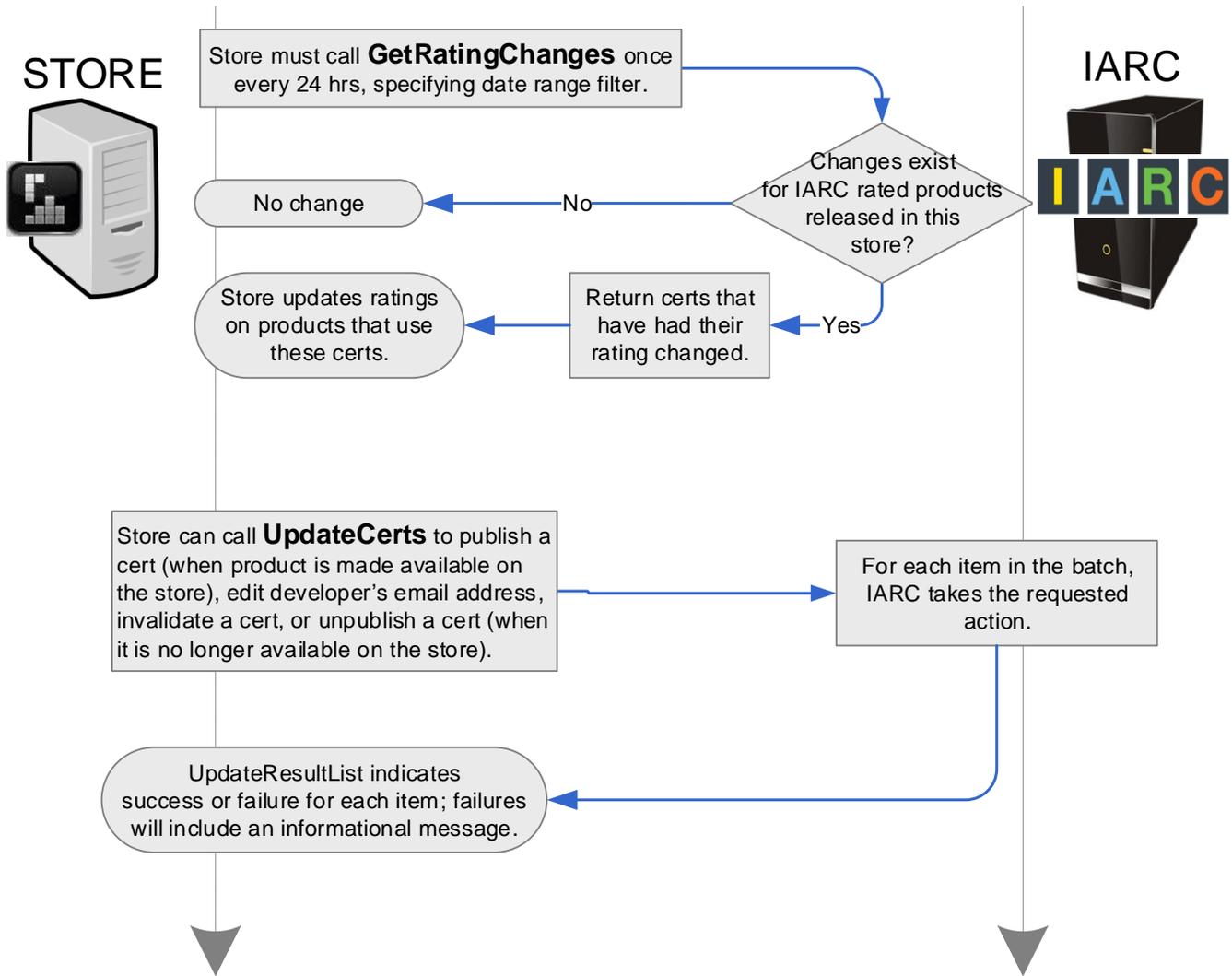
### 3.7 Product Rating Flow: IARC Hosted Configuration



### 3.8 Product Rating Flow: Store Hosted Configuration



### 3.9 Background Processing Web Services



## 4 IARC Website

Used In Configuration: IARC Hosted

### 4.1 Overview

In the IARC Hosted configuration, the store will send the developer to the IARC Website to complete the questionnaire and obtain ratings.

At a high level, the flow is as follows:

1. The store sends the user to IARC, posting only three parameters: storeID, StoreRequestID, and languageID.

2. IARC allows the user to use the system.
3. Based on the responses provided to the questionnaire, IARC generates a rating certificate (age ratings, descriptors, and interactive elements).
4. IARC calls a web service provided by the store, sending the rating certificate and StoreRequestID, using a trusted URL that was provided during design time.
5. If the store recognizes, and is expecting, the StoreRequestID, the store sends a response that includes all the remaining fields that IARC needs, such as product name, company name, etc.

---

#### 4.1.1 Store Provided Web Service: PutCert

The IARC Website will call this store-provided web service to transmit the IARC certificate. Full details on PutCert are below.

---

#### 4.1.2 Developer Redirect

The store opens a new browser window and loads the IARC Website, passing parameters in the URL. Complete details are below.

---

#### 4.1.3 Questionnaire Page

On the IARC Website, the developer is presented with the questionnaire over a series of pages.

- The entire questionnaire is presented on one page, with no postbacks to the web server until the user is done with the questionnaire.
- Some questions are mutually exclusive, some are not.
- All questions use radio buttons or checkboxes for user input. There is no freetext user input.
- All questions require a response.

---

#### 4.1.4 Help Content

For some questions and responses, there is “Help” content that explains in detail the meaning of the question or response and provides examples. The developer clicks a “Help” link, which opens a modal window, which contains tabbed content including example images and videos.

---

#### 4.1.5 Ratings Preview Page

When all questions have been answered by the developer, IARC displays the ratings that were generated. The developer can click “Back” if they would like to adjust their answers.

---

#### 4.1.6 Ratings Confirmation Page

If the developer does not wish to adjust their answers, they click “Next”, at which point IARC calls the store’s PutCert service to receive all necessary attributes of the product (e.g. ProductName, DeveloperEmailAddress, etc.). If the PutCert response indicates success, the developer is given the final confirmation page which includes their newly generated Certificate ID.

---

#### 4.1.7 Email to Developer

Since at this point in time, the developer is normally still in the process of submitting or updating their product at the store, they are normally not ready to publish the product yet, in which case the store will set Publish=false

in the PutCert response. Later when the developer is ready to publish, the store will call UpdateCerts with Action=Publish, which among other things will trigger IARC to email the certificate to the developer.

The store does have the option to set Publish=true in the PutCert response, in which case IARC emails the certificate immediately.

The email displays all the same ratings information that was displayed on the Confirmation Page; it includes a link to a permanent certificate page where the developer can view and print that certificate; and it includes a link to the Rating Check Request page where the developer can request that one or more rating boards conduct a manual review of the auto-generated IARC rating, and where the developer can type comments explaining why they feel the rating should be reviewed.

#### 4.1.8 Localization

The IARC hosted questionnaire is available in 23 languages. IARC LanguageID fields use:

- A two-letter ISO 639-1 language designator
- Followed by a two-letter capitalized ISO 3166-1 alpha-2 region designator

Supported languages are as follows:

ar-SA	Arabic (Saudi Arabia)	ja-JP	Japanese
cs-CZ	Czech	ko-KR	Korean
da-DK	Danish	nb-NO	Norwegian (Norway)
de-DE	German (Standard)	nl-NL	Dutch (Standard)
el-GR	Greek	pl-PL	Polish
en	English (USA)	pt-BR	Portuguese (Brazil)
es-ES	Spanish (Spain)	pt-PT	Portuguese (Portugal)
fi-FI	Finnish	ru-RU	Russian
fr-FR	French (France)	sv-SE	Swedish (Sweden)
he-IL	Hebrew	zh-CN	Chinese (PRC)
hu-HU	Hungarian	zh-TW	Chinese (Taiwan)
it-IT	Italian		

Note: Our “en” texts are American English (i.e. en-US); we are using “en” for reasons of backwards compability and interoperability.

#### 4.2 Developer Redirect

1. During design time, the store will provide to IARC a callback URL. This is part of a pre-established trusted relationship.
2. Developer workflow begins: the store authenticates the developer via username and password or multi-factor authentication.
3. The store sends the developer to the IARC hosted questionnaire, in what is conceptually an SSO-style identity assertion, i.e. IARC does not require the developer to re-authenticate. Normally this process would require a single sign-on solution such as SAML; however, given the specifics of this system integration we can achieve the same level of security without the overhead of the SAML protocol, X.509 certificates, and digital signing/verification.

4. To send the developer to IARC, the store opens a new window, loads a URL to be provided during development phase, and includes the the following fields in the querystring:
  - a. StoreID – IARC assigned GUID for the store.
  - b. StoreRequestID – The store creates this one time request ID, unique for this request.
  - c. LanguageID – The developer’s preferred language, for the questionnaire UI. See Localization section above for specifics on the ISO language codes.

### 4.3 PutCert Web Service

#### 4.3.1 Workflow

1. Upon receiving the developer redirect form post, the first thing IARC does is check to see if we already have the StoreRequestID on file. If we do not, we proceed to step 2 below. If we do:
  - a. We call PutCert, passing the certificate that we already have for this StoreRequestID. We set the field AlreadyExists=true so that the store knows a duplicate call occurred.
    - i. If the store intentionally made a duplicate call, the store can proceed to ingest this certificate.
    - ii. If the store did not intentionally make a duplicate call, then suspicious activity may have occurred, e.g. a replay attack. The store would set StatusCode=NoDuplicate in the response, IARC would log this as a red flag, and can then take additional security measures.
2. The developer completes the questionnaire; IARC generates the ratings, then calls PutCert them to the store.
3. StoreRequestID, which was passed to IARC over HTTPS, acts as a one-time password, it authenticates this web service request.
4. If at any time the store receives a StoreRequestID that it was not expecting to receive, suspicious activity may have occurred; the store rejects the certificate information and returns StatusCode=InvalidRequest.
5. The store then uses StoreRequestID to update the ongoing session with the developer with the new rating information, e.g. refresh the storefront page and display the new certificate to that developer.
6. To prevent replay attacks, StoreRequestID should only be considered valid for a reasonable time window. This duration, which can be decided during the development phase, needs to be long enough for the developer to fill out the entire questionnaire, but not much longer; we suggest one hour.

#### 4.3.2 PutCert Request

Field	Type	Nullable	Description
StoreRequestID	String	N	Store generated unique ID for this transaction. See above for all usage details.
CertID	String (GUID)	N	IARC assigned globally unique ID for the cert.
AlreadyExists	Boolean	N	The store posts a form to IARC, including StoreRequestID as a form field; IARC checks to

			see if we already have that StoreRequestID, and if we do, we set AlreadyExists=true, and we return the RatingList that we already have.
RatingList	Array	N	Array of Rating objects.

### Rating Object

Field	Type	Nullable	Description
RatingID	String (GUID)	N	IARC assigned globally unique ID for this rating decision by this rating authority. When a rating authority modifies a rating, that creates a new RatingID value.
RatingAuthorityID	Int32	N	IARC unique ID for the rating authority.
RatingAuthorityShortText	String	N	E.g. ESRB, PEGI, USK, ClassInd, ACB
RatingLogicVersion	String	N	Version of rating logic used in the Offline Ratings Calculator, hosted on the servers of the store. Example: "5.2"
AgeRatingID	Int32	N	IARC unique ID for the age rating provided by this authority.
AgeRatingText	String	N	The non-localized display text, e.g. "Everyone"
AgeRatingShortText	String	N	The short text representation of an age rating. Unique within a rating board, no spaces, can be used as ID or for user display. Examples: ESRB: E, E10+, T, M, AO ClassInd: L, 10, 12, 14, 16, 18
NumericLevel	Int32	N	The ranking of the age rating within a rating authority's set of age ratings. E.g. ESRB has 5 age ratings; the lowest is E, which has NumericLevel 1, and the highest is AO, with NumericLevel 5.
DescriptorList	Array	N	Array of 0 or more Descriptor objects, for this certificate under this rating authority.
InteractiveElementList	Array	N	Array of 0 or more InteractiveElement objects.

### Descriptor Object

Field	Type	Nullable	Description
DescriptorID	Int32	N	IARC unique ID for the descriptor.
DescriptorText	String	N	The value will be a meaningful string ID to be used for looking up localized display text (we call this a

			messageID within IARC). That display text will be a content descriptor assigned by a rating authority, e.g. ESRB might assign “Blood and Gore” or “Crude Humor”.
DisplayOrder	Int32	N	When multiple descriptors are present for a given rating authority, they are in a specific order according to the rules of that rating authority, and should be shown in the UI in that order.

### InteractiveElement Object

Field	Type	Nullable	Description
InteractiveElementID	Int32	N	IARC unique ID for the interactive element.
InteractiveElementText	String	N	Value will be an ID to look up display text. IDs: <ul style="list-style-type: none"> <li>• IE_UsersInteract</li> <li>• IE_SharesInfo</li> <li>• IE_SharesLocation</li> <li>• IE_DigitalPurchases</li> <li>• IE_UnrestrictedInternet</li> </ul> Display text in English: <ul style="list-style-type: none"> <li>• Users Interact</li> <li>• Shares Info</li> <li>• Shares Location</li> <li>• Digital Purchases</li> <li>• Unrestricted Internet</li> </ul>

### 4.3.3 PutCert Response

Field	Type	Nullable	Description
StatusCode	String (enum)	N	Code indicating the end result of the developer transaction, as discussed above. Possible values: <ul style="list-style-type: none"> <li>• Success</li> <li>• Failure</li> <li>• NoDuplicate</li> <li>• InvalidRequest</li> </ul>
Publish	Boolean	N	If the product will become available on the storefront as of this transaction, the store will set this to true. If the developer will publish the product on the storefront at a later date,

			the store will set this to false.
ProductName	String	N	Name of the product that has been rated.
StoreProductID	String	N	Store assigned ID for the product. Among other things, this allows IARC to connect multiple certs to the same product if needed.
StoreProductUrl	String	N	URL to the product page on the store for this product.
StoreDeveloperID	String	N	Store assigned ID for the developer.
CompanyName	String	N	Developer company name.
EmailAddress	String	N	Developer's email address, for the purpose of IARC communications.
LocalizedTitles	Object Array	Y	<p>Array of LocalizedTitle objects. If a store allows the developer to provide multiple titles for the product, each in a different language, those will be provided here.</p> <p>In some scenarios, the ProductName field may not be the product name that is viewable on the store, and in those cases the rating authority can look for product on the store using the LocalizedTitle.</p>
CountryCodes	String Array	Y	<p>A list of country codes that specify where this product is available. These are two-letter capitalized ISO 3166-1 alpha-2 region designators.</p> <p>This informs the rating authorities which countries they will be able to test the product in.</p>
Platforms	String Array	Y	A list of the platforms for which this product is available. This is to be human readable display text, e.g. it informs the rating authorities which platforms to use when testing the product.

### LocalizedTitle Object

Field	Type	Nullable	Description
Title	String (UTF-8)	N	Localized UTF-8 display text.
LanguageID	String	N	<p>Same as specified elsewhere in this doc:</p> <ul style="list-style-type: none"> <li>- A two-letter ISO 639-1 language designator</li> <li>- Followed by a two-letter capitalized ISO</li> </ul>

			3166-1 alpha-2 region designator E.g. zh-CN, pt-BR, fr-FR.
--	--	--	---

## 5 IARC Web Services

### 5.1 Shared Behavior

#### 5.1.1 Overview

IARC provides a set of JSON web services over HTTPS. (Attempts to access via HTTP are prevented.) For every service, the client POSTs custom JSON objects defined in this spec. The web services do not use a RESTful style and do not expose CRUD operations, they carry out application logic operations. Authentication is handled in the HTTP headers. Content-Type and Accept headers should be set to application/json.

#### 5.1.2 Security

##### Authentication

Authenticating the client will be handled via name and password provided at the message level: StoreID and StorePassword (provided to the store during development phase) will be passed via HTTP headers. Authenticating the IARC server will be via an SSL certificate.

Field	Type	Nullable	Description
StoreID	String	N	IARC provided ID for a client who has been granted access to call this service.
StorePassword	String	N	IARC provided password for the the store.

##### Confidentiality

Transfer encryption is required because the messages will contain a small amount of personally identifiable information (PII): the rating authorities require that IARC collect a contact email address for the developer. This encryption will take place at the transport level, via SSL.

##### Authorization

Access control is based on the StoreID passed within the message. When a store partners with IARC, a StoreID is created and added to the access control list for the store-facing IARC web services.

#### 5.1.3 Localization

IARC supports 23 languages. For all IARC generated display text, IARC will supply to the store the translations that should be used for those 23 languages. This includes descriptors, interactive elements, question categories, questions, responses, and help content.

Note that age ratings and rating board names are not localized. The rating authorities have chosen to keep these in the one representation they have supplied. The store will display an icon for the age rating (e.g. E10, L), and

additionally display a non-localized descriptive text field (e.g. “Everyone 10+”, “Livre”). The rating board names supplied are in their native language, e.g. USK is “Unterhaltungssoftware Selbstkontrolle”.

IARC LanguageID fields use:

- A two-letter ISO 639-1 language designator
- Followed by a two-letter capitalized ISO 3166-1 alpha-2 region designator

Supported languages are as follows:

ar-SA	Arabic (Saudi Arabia)	ja-JP	Japanese
cs-CZ	Czech	ko-KR	Korean
da-DK	Danish	nb-NO	Norwegian (Norway)
de-DE	German (Standard)	nl-NL	Dutch (Standard)
el-GR	Greek	pl-PL	Polish
en	English (USA)	pt-BR	Portuguese (Brazil)
es-ES	Spanish (Spain)	pt-PT	Portuguese (Portugal)
fi-FI	Finnish	ru-RU	Russian
fr-FR	French (France)	sv-SE	Swedish (Sweden)
he-IL	Hebrew	zh-CN	Chinese (PRC)
hu-HU	Hungarian	zh-TW	Chinese (Taiwan)
it-IT	Italian		

Note: Our “en” texts are American English (i.e. en-US); we are using “en” for reasons of backwards compability and interoperability.

#### 5.1.4 Date Format and Precision

All dates in IARC web service responses are:

- ISO 8601 extended format
- UTC timezone
- Microsecond resolution
- yyyy-MM-ddTHH:mm:ss.mmmmmmZ

All dates passed in the web service requests *must be UTC*, and can be any of the following formats:

yyyy-MM-dd

yyyy-MM-ddTHH:mm:ss

yyyy-MM-ddTHH:mm:ss.SSSZ

yyyy-MM-ddTHH:mm:ss.mmmmmmZ

## 5.2 LoadQuestionnaire

**Used In Configuration: Store Hosted**

### 5.2.1 Business Logic

- This service returns a complete list of all the questions in the questionnaire.
- Every question is bundled with its set of possible responses.
- Every question has a question category attribute, e.g. Violence, Sexuality, Controlled Substances.
- Some questions and some responses are bundled with Help HTML content, which can contain images or videos (embedded YouTube videos), and in some cases have tabbed content (i.e. help objects that contain help objects).
- This service also returns a list of Triggers, which indicate which questions should be shown as followup questions:
  - A Trigger is simply a ResponseInstanceID paired with one or more FollowupQuestionInstanceIDs.
  - Every time the developer provides a response, that ResponseInstanceID should be looked up in the TriggerList, and if found, the associated followup question should then be displayed.
- All questions are multiple-choice (no freetext user input).

---

### 5.2.2 Localization

- All display text is represented by what we call a messageID
- The IARC team also provides the Message table, as a pipe delimited file. This table has three columns: messageID, languageID, and displayText.
- The store uses the messageID, plus the desired languageID, to look up the displayText for that language.

Question and response text occasionally contain some span or div tags with IARC-defined CSS class IDs on them. These can be ignored by the store, or the store can define their own styles for those classes. Help text very often contains HTML, e.g. image tags and embedded videos (via YouTube), which should not be ignored.

---

### 5.2.3 JSON Example: Child Questions

A question can have a set of child questions. Currently in questionnaire version 6.3, there are only two instances of this, and both ask the user to “Please check all that apply”. (Note that the concept of child question is not related to the concept of followup question.)

LANGUAGE

Does the game contain any potentially offensive language? Please note that this question does *not* refer to user generated content.

Yes  No

ResponseCode

ResponseInstanceList, MutuallyExclusiveResponses=true

(Note the italicized word “not”; this is an example of a div tag with an IARC-defined CSS class.)

The above is taken from the following JSON:

```
{
  "QuestionInstanceID": 892,
  "QuestionCode": "Language_Game",
  "ReplacesQuestionID": null,
  "CanUsePreviousResponse": null,
  "IsRoot": false,
  "QuestionSortOrder": 121,
  "QuestionCategoryCode": "Language_Category",
  "MutuallyExclusiveResponses": true,
  "ResponseInstanceList": [
    {
      "ResponseInstanceID": 2107,
      "ResponseCode": "Yes",
      "ResponseSortOrder": 0,
      "TextForSummary": null
    },
    {
      "ResponseInstanceID": 2108,
      "ResponseCode": "No",
      "ResponseSortOrder": 1,
      "TextForSummary": null
    }
  ],
  "ChildQuestionList": [],
  "Help": null
},
```

Above, the “Yes” response has ResponseInstanceID 2107, which in the TriggerList points to FollowupQuestionID 893:

```
{
  "ResponseInstanceID": 2107,
  "FollowupQuestionIDList": [
    893
  ]
},
```

QuestionInstanceID 893 is then shown to the user; it is a parent question that has 3 child questions, highlighted in yellow below. A parent question does not have responses of its own; it has the child questions in place of having responses.

## LANGUAGE

Does the game contain any potentially offensive language? Please note that this question does *not* refer to user generated content. ✓

Yes  No

Which of the following potentially offensive words or categories of words are contained in the game? Please check all that apply

Mild swearing and/or language that could be considered offensive

Rarely  Often  Never

Discriminatory language (against race, religion, sex, etc...)

Rarely  Often  Never

Sexual expletives

Rarely  Often  Never

JSON for the above is shown on the next page.

```

{
  "QuestionInstanceID": 893,
  "QuestionCode": "L_Presentation_Game",
  "ReplacesQuestionID": null,
  "CanUsePreviousResponse": null,
  "IsRoot": false,
  "QuestionSortOrder": 122,
  "QuestionCategoryCode": "Language_Category",
  "MutuallyExclusiveResponses": false,
  "ResponseInstanceList": [],
  "ChildQuestionList": [
    {
      "QuestionInstanceID": 940,
      "QuestionCode": "Swearing",
      "ReplacesQuestionID": null,
      "CanUsePreviousResponse": null,
      "IsRoot": false,
      "QuestionSortOrder": 124,
      "QuestionCategoryCode": "Language_Category",
      "MutuallyExclusiveResponses": true,
      "ResponseInstanceList": [
        {
          "ResponseInstanceID": 2210,
          "ResponseCode": "Rarely",
          "ResponseSortOrder": 0,
          "TextForSummary": "Summary_SwearingRarely"
        },
        {
          "ResponseInstanceID": 2211,
          "ResponseCode": "Often",
          "ResponseSortOrder": 1,
          "TextForSummary": "Summary_SwearingOften"
        },
        {
          "ResponseInstanceID": 2281,
          "ResponseCode": "Never",
          "ResponseSortOrder": 2,
          "TextForSummary": null
        }
      ]
    },
    "ChildQuestionList": [],
    "Help": null
  ],
},

```

## 5.2.4 JSON Example: Help Content

### VIOLENCE AGAINST HUMANS

Does the game contain violence or implied violence against humans? 

Yes  No

The JSON values for the question above look the same as for the Language question in the first example above, i.e. simple yes/no question:

```
{
  "QuestionInstanceID": 828,
  "QuestionCode": "HumanViolence_Game",
  "ReplacesQuestionID": null,
  "CanUsePreviousResponse": null,
  "IsRoot": false,
  "QuestionSortOrder": 57,
  "QuestionCategoryCode": "VAH_Category",
  "MutuallyExclusiveResponses": true,
  "ResponseInstanceList": [
    {
      "ResponseInstanceID": 1940,
      "ResponseCode": "Yes",
      "ResponseSortOrder": 0,
      "TextForSummary": "Summary_HumanViolence_Game"
    },
    {
      "ResponseInstanceID": 1941,
      "ResponseCode": "No",
      "ResponseSortOrder": 1,
      "TextForSummary": null
    }
  ],
  "ChildQuestionList": [],
}
```



The JSON above gets rendered in the UI as follows:

The screenshot shows a help dialog box titled "Violence Against Humans" with a close button (X) in the top right corner. The dialog contains the following text: "The characters must look like humans. If a character looks like a human, it should be treated as a human even if they are characters that are clad in armor or obscured in some other way, but understood to also be considered human. Implied violence against humans, such as the shooting of vehicles understood to have humans inside, would also be considered violence against humans." Below the text are two radio buttons: "Humans" (selected) and "Implied Humans". Underneath the radio buttons is the text "Example screenshots of characters that look like humans:" followed by four images in a 2x2 grid. The top-left image shows a character with a large, bloody wound on their chest. The top-right image shows a character in a dark jacket in a post-apocalyptic setting. The bottom-left image shows a character with a long orange beard and blue armor. The bottom-right image shows a character in a sailor outfit in a lobby. A "Close" button is located at the bottom right of the dialog. Several orange callout boxes with labels point to specific elements: "HelpTitleCode" points to the title; "HelpTextCode" points to the main text; "HelpList" points to the radio buttons; "HelpImageList" points to the grid of images.

HelpTitleCode

Violence Against Humans

HelpTextCode

The characters must look like humans. If a character looks like a human, it should be treated as a human even if they are characters that are clad in armor or obscured in some other way, but understood to also be considered human. Implied violence against humans, such as the shooting of vehicles understood to have humans inside, would also be considered violence against humans.

HelpList

Humans  Implied Humans

HelpTitleCode

HelpTextCode

Example screenshots of characters that look like humans:



HelpImageList

Close

### 5.2.5 Call Frequency

The IARC questionnaire will not change more often than every 6 months. Any change to the questionnaire is coordinated closely, far in advance, with all stores that use IARC.

1. The store will be notified when the new version is available
2. The call to this service will then be triggered manually
3. IARC will provide to the store a new Message table which contains changes for the new version
4. If the new version entails new or different help image files, IARC will deliver those as well
5. The store will then have time to try out the new questionnaire and see if they need or want to make changes in their UI for this new version, prior to the store promoting it to production.

### 5.2.6 LoadQuestionnaire Request

Field	Type	Nullable	Description
QuestionnaireVersion	String	N	Major and minor version of the questionnaire, e.g. "6.3".  All rating authorities are in agreement to use the same single questionnaire for all products across all stores. When changes are made to this shared questionnaire, a new version number will be used.

### 5.2.7 LoadQuestionnaire Response

Field	Type	Nullable	Description
QuestionnaireVersion	String	N	Version of the questionnaire that was requested, e.g. "6.3".
LatestVersion	String	N	Version of the most current IARC questionnaire. If IARC has moved to version 6.4, this will show "6.4". If the store had requested 6.3, the store will then know to make a subsequent call for 6.4.
QuestionInstanceList	Array	Y	Array of QuestionInstance objects; the complete list of questions for the questionnaire. Each question comes paired with the set of possible responses for that question, as well as occasional help text that explains the question in more detail.  Will only be null if the version requested

			does not exist.
TriggerList	Array	Y	<p>Array of Trigger objects. This is the logic for displaying follow-up questions: a list of logic expressions, each with a list of one or more follow-up questions; when a logic expression evaluates to true, the associated follow-up question(s) should be displayed.</p> <p>Will only be null if the version requested does not exist.</p>

### QuestionInstance Object

Field	Type	Nullable	Description
QuestionInstanceID	Int32	N	Internal IARC ID for the unique instance of this question in this questionnaire. Unique within a questionnaire, but will be used across multiple questionnaire versions.
QuestionCode	String	N	This is a messageID to be used for looking up the question display text in the Message table for the desired languageID.
IsRoot	Boolean	N	This will be true for the first set of questions to be asked; i.e. the questions to be shown on page one of the questionnaire are called root questions. Currently there is only one root question, asking the developer to pick a product category (aka “app type”) for their product.
QuestionSortOrder	Int32	N	Indicates sort order of questions, for UI. QuestionInstanceList should be sorted by this field. FollowupQuestionIDList should also be sorted by this field.
QuestionCategoryCode	String	Y	This is a messageID to be used for looking up the question category display text in the Message table for the desired languageID. Examples of question categories are Violence, Language, Controlled Substances, etc.
MutuallyExclusiveResponses	Boolean	Y	When true, show the response list with mutually exclusive inputs e.g. radio buttons. When false, show the response list with non-mutually exclusive UI e.g. checkboxes.
ResponseInstanceList	Array	Y	Array containing 2 or more ResponseInstance objects, i.e. possible responses to the question.

ChildQuestionList	Array	Y	When this is a parent question, this will list its children as a nested question list, i.e. a list of QuestionInstance objects (currently these will not have children themselves).
Help	Object	Y	Each help message is intended to clarify the meaning of a specific question on the questionnaire, and should be offered to the user next to that question. This field contains URL-encoded HTML. Not every Question has Help, so this field is nullable.

### ResponseInstance Object

Field	Type	Nullable	Description
ResponseInstanceID	Int32	N	Every QuestionInstance has a list of ResponseInstances, i.e. the list of responses the user can choose for this question.
ResponseCode	String	N	This is a messageID to be used for looking up the response display text in the Message table for the desired languageID.
ResponseSortOrder	Int32	N	Indicates sort order for UI. ResponseInstanceList should be sorted by this field.
TextForSummary	String	N	<p>This is a messageID to be used for looking up the summary display text in the Message table for the desired languageID.</p> <p>When the developer is finished with the questionnaire, they can be shown a summary of the information they provided. This field contains the text to be displayed in that summary, if this response was selected by the developer.</p> <p>Example: If the developer answers “Yes” to the question “Is the use of drugs related to incentives or rewards?”, the summary page would include the TextForSummary “Use of illegal drugs includes incentives”.</p>
Help	Object	Y	Helps can be present on both questions and responses.

### Trigger Object

Field	Type	Nullable	Description
-------	------	----------	-------------

ResponseInstanceID	Int32	N	IARC internal ID for a response, unique within the questionnaire.
FollowupQuestionIDList	Array	N	Array of one or more questionInstanceIDs (type=Int32) for followup questions that should be shown if the developer chose the above ResponseInstanceID.

### Help Object

Field	Type	Nullable	Description
HelpTitleCode	String	Y	A messageID to be used for looking up the help title display text in the Message table for the desired languageID.
HelpTextCode	String	Y	A messageID to be used for looking up the help display text in the Message table for the desired languageID.  XML compliant HTML text. HTML will be strictly structural, e.g. following tags: p, h1, strong.
HelpImageList	Array	Y	List of HelpImage elements
HelpVideoList	Array	Y	List of HelpVideo elements
HelpList	Array	Y	List of one or more Help elements. I.e. a Help can contain a list of Helps; these are intended to be shown as tabbed content, with each child help shown in its own tab.
SortOrder	Int32	Y	Not required when this Help is not part of a list.

### HelpImage Object

Field	Type	Nullable	Description
Filename	String	N	Name for the image file. These files will be delivered to the store to be hosted by the store.
SortOrder	Int32	N	Sort order for when there are multiple HelpImages within a Help.
Width	Int32	Y	Width of the image file in pixels.
Height	Int32	Y	Height of the image file in pixels.

### HelpVideo Object

Field	Type	Nullable	Description
VideoUrl	String	N	URL of the video. Currently these are all YouTube URLs. Example: <a href="https://www.youtube.com/embed/lhDaKJ3Z0ms?rel=0&amp;showinfo=0">https://www.youtube.com/embed/lhDaKJ3Z0ms?rel=0&amp;showinfo=0</a>
SortOrder	Int32	N	Sorting for when there are multiple videos on the same help content.

## 5.3 SearchCerts

Used In Configurations: IARC Hosted and Store Hosted

### 5.3.1 Business Logic

Searches IARC for certs that already exist.

Use case scenario: if the developer submitted the product at Store A, they are not required to complete the questionnaire again to release this same product on Store B; the store prompts the developer to enter the CertID, and then uses this service to obtain the pre-existing rating data for that submission.

Call Frequency: Whenever a developer chooses to locate an existing cert.

### 5.3.2 SearchCerts Request

Field	Type	Nullable	Description
CertID	String (GUID)	N	Globally unique ID for this cert.

### 5.3.3 SearchCerts Response

#### Cert Object

Field	Type	Nullable	Description
MatchFound	Boolean	N	Boolean indicating whether or not a match was found for the CertID in the request. The next two fields will be null when this is false.
CertID	String (GUID)	Y	Globally unique ID for the cert.
ProductName	String	Y	This is the name that the developer provided for the product at the time that they originally created this cert, at whichever store that was.
DateCreated	DateTime	Y	Date the cert was created at that store. Again, to help the developer know which cert this is.
ResponseInstanceIDList	Int32 Array	Y	Unordered array of Int32 ResponseInstanceIDs, i.e. all the responses that the developer gave for the

			questionnaire.
QuestionnaireVersion	String	Y	<p>The version of the questionnaire that the developer took when they provided the responses shown in ResponseInstanceIdList.</p> <p>So if the store supports that version, the store can use the response ID list to prepopulate the questionnaire if the developer wants to retake it; but if the store does not support that version of the questionnaire, that's not an option.</p>
RatingList	Array	Y	Array of Rating objects. The set of ratings that the store generated using the Offline Ratings Calculator.

### Rating Object

Field	Type	Nullable	Description
RatingID	String (GUID)	N	<p>IARC assigned globally unique ID for this rating decision by this rating authority. When a rating authority modifies a rating tied to a cert, that creates a new RatingID value.</p> <ul style="list-style-type: none"> <li>Stores that host the questionnaire will ignore RatingID because ORC does not generate this field.</li> <li>Stores that use the IARC Hosted Configuration will receive RatingID in the PutCert service, and can use it in all interactions.</li> </ul>
RatingAuthorityID	Int32	N	IARC unique ID for the rating authority.
RatingAuthorityShortText	String	N	E.g. ESRB, PEGI, USK, ClassInd, ACB
RatingLogicVersion	String	N	Version of rating logic used in the Offline Ratings Calculator, hosted on the servers of the store. Example: "5.2"
AgeRatingID	Int32	N	IARC unique ID for the age rating provided by this authority.
AgeRatingText	String	N	The non-localized display text, e.g. "Everyone"
AgeRatingShortText	String	N	<p>The short text representation of an age rating. Unique within a rating board, no spaces, can be used as ID or for user display.</p> <p>Examples: ESRB: E, E10+, T, M, AO</p>

			ClassInd: L, 10, 12, 14, 16, 18
DescriptorList	Array	N	Array of 0 or more Descriptor objects, for this certificate under this rating authority.
InteractiveElementList	Array	N	Array of 0 or more InteractiveElement objects.

### Descriptor Object

Field	Type	Nullable	Description
DescriptorID	Int32	N	IARC unique ID for the descriptor.
DescriptorText	String	N	The value will be a meaningful string ID to be used for looking up localized display text (we call this a messageID within IARC). That display text will be a content descriptor assigned by a rating authority, e.g. ESRB might assign “Blood and Gore” or “Crude Humor”.
DisplayOrder	Int32	N	When multiple descriptors are present for a given rating authority, they are in a specific order according to the rules of that rating authority, and should be shown in the UI in that order.

### InteractiveElement Object

Field	Type	Nullable	Description
InteractiveElementID	Int32	N	IARC unique ID for the interactive element.
InteractiveElementText	String	N	Value will be an ID to look up display text. IDs: <ul style="list-style-type: none"> <li>• IE_UsersInteract</li> <li>• IE_SharesInfo</li> <li>• IE_SharesLocation</li> <li>• IE_DigitalPurchases</li> <li>• IE_UnrestrictedInternet</li> </ul> Display text in English: <ul style="list-style-type: none"> <li>• Users Interact</li> <li>• Shares Info</li> <li>• Shares Location</li> <li>• Digital Purchases</li> <li>• Unrestricted Internet</li> </ul>

## 5.4 AttachToCert

Used In Configurations: IARC Hosted and Store Hosted

### 5.4.1 Business Logic

Used for an existing cert created by a different store, a scenario referred to as ratings portability, wherein the developer has already taken the questionnaire at Store A to create a cert, and then wants to use that cert at Store B.

- Store B will prompt the developer to enter the certID, and call SearchCerts to find that cert.
- Store B will present the ratings information, along with other fields returned by SearchCerts, to the developer and ask if this is the cert they want to use.
- When the developer confirms it, the store will call AttachToCert, supplying fields described below.

If the provided StoreProductID is already tied to a different CertID for this store, i.e. the product already has a cert, IARC will set that linking to “inactive” and point the provided StoreProductID to the provided CertID (i.e. we will swap from the old cert to the new cert).

If the provided StoreProductID is already tied to the provided CertID for this store, IARC will return an error message.

The default behavior of the service is that a store can only attach one StoreProductID to a given CertID. However, a store can request the Multiple Product IDs ability, if the store uses multiple StoreProductIDs for a given product.

#### Default Behavior

- If the provided CertID is already tied to a different StoreProductID for this store, i.e. the cert is already “in use”, IARC will return an error message.

#### Optional: Multiple Product IDs Behavior

- If the provided CertID is already tied to a different StoreProductID for this store, i.e. the cert is already “in use”, IARC will not consider this an error, the store is free to tie any number of StoreProductIDs to a given CertID.
- If this option is requested, IARC implements this as a persistent configuration setting, i.e. the parameters passed to the service are the same whether or not this option is enabled.

### 5.4.2 AttachToCert Request

Field	Type	Nullable	Description
CertID	String (GUID)	N	Globally unique ID for the certificate
DeveloperEmail	String	N	Email address that the developer has provided for the purpose of IARC communications.  IARC will treat this email address as the current contact address not just for this cert but <i>for the developer</i> : this means IARC will look at all certs for the StoreDeveloperID provided in this request, and will overwrite existing email address with this provided email address.

StoreProductID	String	N	Store assigned unique ID for the product.
StoreProductURL	String	N	The URL to the product page on the storefront.
StoreDeveloperID	String	N	Store-assigned unique ID for the developer.
CompanyName	String	N	Name of the developer company as it appears on this store.
ProductName	String	N	Name of the product as it appears on this store.
Publish	Boolean	N	Set to true if this app has been made available for sale/download on this store. Setting this to false means that when the app does become available, the store will call UpdateCerts with Action=Publish.
DisplayText	String (10 char limit)	Y	This exists for disambiguation when a cert is in use on multiple products at the same store. The store should supply a short string indicating the regions where this product will be published. This way, in the UI, when the name of the product shows up multiple times for a given cert, IARC's UI can show the region next to each one to explain the difference between them.
LocalizedTitles	Object Array	Y	<p>Array of LocalizedTitle objects. If a store allows the developer to provide multiple titles for the product, each in a different language, those will be provided here.</p> <p>In some scenarios, the ProductName field may not be the product name that is viewable on the store, and in those cases the rating authority can look for product on the store using the LocalizedTitle.</p>
CountryCodes	String Array	Y	<p>A list of country codes that specify where this product is available. These are two-letter capitalized ISO 3166-1 alpha-2 region designators.</p> <p>This informs the rating authorities which countries they will be able to test the product in.</p>
Platforms	String Array	Y	A list of the platforms for which this product is available. This is to be human readable display text, e.g. it informs the rating authorities which platforms to use when testing the product.

#### LocalizedTitle Object

Field	Type	Nullable	Description
Title	String (UTF-8)	N	Localized UTF-8 display text.

LanguageID	String	N	Same as specified elsewhere in this doc: <ul style="list-style-type: none"> <li>- A two-letter ISO 639-1 language designator</li> <li>- Followed by a two-letter capitalized ISO 3166-1 alpha-2 region designator</li> </ul> E.g. zh-CN, pt-BR, fr-FR.
------------	--------	---	--

### 5.4.3 AttachToCert Response

Field	Type	Nullable	Description
ResultCode	String	N	Value is either Success or Failure.
ErrorID	String (GUID)	Y	When ResultCode=Failure, this will be a globally unique ID that the IARC team can use for locating the event in the system logs.
ErrorMessage	String	Y	When ResultCode=Failure, this will contain human readable descriptions of the error condition. Calling this service declares that the store would like to start using this certID at this store. <ul style="list-style-type: none"> <li>• If the service finds that the certID is already in use at that store, it will return an error.</li> <li>• If certID doesn't exist, it will return an error.</li> </ul>

## 5.5 CreateCerts

**Used In Configurations: Store Hosted**

### 5.5.1 Business Logic

Passes submission data to IARC (product data, ORC ratings and questionnaire responses) and then returns a newly created certificate. Supports batch operations, i.e. the request contains a list of submissions.

The store will call this service upon release of the product. The rating authorities will consider the release date of the product to be the date that this service is called.

- Upon product release, the store will maintain a queue of submissions to be sent to IARC via CreateCerts.
- Call frequency: The store will call CreateCerts when the queue length reaches a 100, but not more often than 60 seconds apart.
- For each submission sent in the request, IARC will:
  - Generate ratings information based on the responses provided in this service call
  - Return the newly created CertID and ratings
  - Send an email to the developer, with link to their official rating certificate. This web service accepts a language code in the request, and generates this email in the requested language.

### 5.5.2 CreateCerts Request

Field	Type	Nullable	Description
SubmissionList	Object Array	N	Array of one or more Submission objects, i.e., a list of submissions, created locally by the store using Offline Ratings Calculator, to be inserted at IARC.

### Submission Object

Field	Type	Nullable	Description
StoreRequestID	String	Y	An ID for this certificate request that is unique within the store. IARC will include this in the response, for request tracking purposes.
Publish	Boolean	N	A store that uses ORC to generate ratings has the option to call CreateCerts when the product is already released/published, and the store will set this flag to true.  Setting this to false is essentially a commitment to call UpdateCerts at a later date, with Action=Publish, to inform IARC that the cert and the product are being released/published at that time.
ProductName	String	N	Name of the product in this submission.
CompanyName	String	N	Name of company that developed the app.
ORCVersion	String	Y	The version of the Offline Ratings Calculator codebase itself, as distinct from the rating logic.  This version will always be of the form <i>major.minor.patch</i> , e.g. 6.1.0.  A change to rating logic requires a change to this codebase, so this major and minor version will always match the rating logic version. But when there is a need for a codebase change without a need for a rating logic change, we will increment the patch version (third number, e.g. 6.1.0 to 6.1.1).  Thus, a rating logic change always pushes up both the rating logic version and the ORC version, but an ORC-only change pushes up only the ORC version.
OfflineGenerationDate	DateTime	N	Timestamp for when the store generated the rating using Offline Ratings Calculator.
StoreProductID	String	Y	Unique ID for the product within the store.
StoreDeveloperID	String	Y	Unique ID for the developer within the store.
StoreProductUrl	String	Y	URL to the product page at the store.
Email	String	N	Email address that the rating authorities can use to

			<p>contact the developer of the product.</p> <p>IARC will treat this email address as the current contact address not just for this cert but <i>for the developer</i>: this means IARC will look at all certs for the StoreDeveloperID provided above, and will overwrite existing email address with this provided email address.</p>
LangCode	String	N	<p>Indicates the language that the developer was using when providing data for this submission. This service creates an email to the developer, and that email will be in the language specified by LangCode if that language is supported by IARC.</p>
QuestionnaireVersion	String	N	<p>Version number for the questionnaire that was answered by the developer when generating this rating data, e.g. “6.1”.</p> <p>Concretely, this refers to the text and help content shown to the user, it does not refer to logic used for generating ratings data upon completion of the questionnaire.</p> <p>This will match RatingLogicVersion most of the time. However, a questionnaire can change without a need for rating logic change, and vice versa.</p> <p>For small changes to either the QuestionnaireVersion or the RatingLogicVersion, the minor version will increment on one without the other (e.g. leading to one being 6.1 while the other is 6.2).</p> <p>For a major update, the major version will increase on both, bringing them back into sync, e.g. from 6.1 and 6.2 to both of them set to 7.0.</p>
ResponseInstanceIDList	Int32 Array	N	<p>Array of one or more Int32 ResponseInstanceIDs, i.e. the response(s) that the developer chose for this question. Some questions have one response (e.g. yes or no), some have multiple (e.g. “please check all that apply”). The order of this list does not matter.</p>
RatingList	Object Array	N	<p>Array of Rating objects. The set of ratings that the store generated using the Offline Ratings Calculator.</p>
LocalizedTitles	Object Array	Y	<p>Array of LocalizedTitle objects. If a store allows the developer to provide multiple titles for the product, each in a different language, those will be provided here.</p> <p>In some scenarios, the ProductName field may not be the product name that is viewable on the store, and in those cases the rating authority can look for</p>

			product on the store using the LocalizedTitle.
CountryCodes	String Array	Y	A list of country codes that specify where this product is available. These are two-letter capitalized ISO 3166-1 alpha-2 region designators. This informs the rating authorities which countries they will be able to test the product in.
Platforms	String Array	Y	A list of the platforms for which this product is available. This is to be human readable display text, e.g. it informs the rating authorities which platforms to use when testing the product.

### LocalizedTitle Object

Field	Type	Nullable	Description
Title	String (UTF-8)	N	Localized UTF-8 display text.
LanguageID	String	N	Same as specified elsewhere in this doc: <ul style="list-style-type: none"> <li>- A two-letter ISO 639-1 language designator</li> <li>- Followed by a two-letter capitalized ISO 3166-1 alpha-2 region designator</li> </ul> E.g. zh-CN, pt-BR, fr-FR.

### Rating Object

Field	Type	Nullable	Description
RatingAuthorityID	Int32	N	IARC unique ID for the rating authority.
RatingLogicVersion	String	N	The version of the rating logic used to generate this rating data. This will only change when the rating authorities need to make a change to rating logic. This version will always be of the form <i>major.minor</i> , e.g. 5.2. Major changes will increment the major version (first number, e.g. 5.2 to 6.0), which historically has not happened more than once a year. Minor changes, such as changing the rating logic for one question, will increment the minor version (second number, e.g. 6.0 to 6.1).
AgeRatingID	Int32	N	IARC unique ID for the age rating provided by this authority.

DescriptorIDList	Int32 Array	N	Array of 0 or more Descriptor objects, for this certificate under this rating authority.
InteractiveElementIDList	Int32 Array	N	Array of 0 or more InteractiveElement objects.

### 5.5.3 CreateCerts Response

The response is an array of Cert objects, i.e. the newly created certificates.

#### Cert Object

Field	Type	Nullable	Description
CertID	String (GUID)	N	Globally unique ID for this cert, which was newly generated during this web service call.
StoreProductID	String	Y	Unique ID for the product within the store.
StoreRequestID	String	Y	An ID for this certificate request that is unique within the store. IARC will include this in the response, for request tracking purposes.
DateCreated	Date	N	Format: yyyy-MM-ddTHH:mm:ss.mmmmmmmZ
AlreadyExists	Boolean	N	If we receive a StoreRequestID that we have received in the past, we will set AlreadyExists to true and load and return the ratings that were generated the first time we received that StoreRequestID.
Result	Object	N	Result of the operation conducted for this record.

#### Result Object

Field	Type	Nullable	Description
ResponseCode	String (Enum)	N	Possible values: Success or Failure
ErrorID	String (GUID)	Y	Empty for successes. Upon failure, this will contain a GUID for the error, which will be logged in the IARC database.
ErrorMessage	String	Y	Empty for successes. For failed validation, this will be a custom error. For unexpected exceptions, this will be the exception message.

### 5.6 GetRatingChanges

Used In Configurations: IARC Hosted and Store Hosted

### 5.6.1 Business Logic

This service allows the store to request all certs that have had their ratings data modified during the time window specified in the request. Such modifications, also called overrides, occur when a rating authority reviews a product after it has been released and finds that its content does not match the disclosure provided by the developer when it was originally submitted.

This service returns ratings that have changed, not certs per se. A cert can be seen as a bucket filled with ratings, one rating per rating authority; if only one rating authority has changed their rating on a given cert, this service returns only the rating for that rating authority. It does not return the ratings that have not changed.

Ratings tied to a cert can be modified at any time; different rating authorities can modify their ratings at different times; and a given rating authority can modify their rating on the cert more than once. If a rating has changed twice for the same rating authority within the requested time window, the service will only return the most recent change; that is, when the store calls this service, providing a date range, IARC searches the rating overrides within that date range, and only the ones that are the *current* rating for each rating authority.

DateCreated fields: The DateCreated fields come from two separate tables, the Cert table and the Rating table. DateCreated on the Cert will naturally never change; but DateCreated on the Rating is useful when looking at rating modifications: it provides one way to see that you are receiving a more recent rating that replaces one you received earlier. However, this service only returns current ratings, so if it's present in this service then it is necessarily the most recent rating. In cases of a rating modification, Rating.DateCreated is the timestamp of that rating modification. The start and end date filter passed in the service request is applied to Rating.DateCreated.

An example scenario where a rating has changed twice (rating1 and rating2):

- Store calls for range T1. IARC returns rating1 (for a given rating authority).
- Store later calls for range T2. IARC returns rating2.
  - If at this time the store calls for T1, rating1 would no longer be returned as rating2 is the latest rating.

Call frequency: The store will call this service once every 24 hrs.

Supports paging for returning large datasets.

Format for StartDate and EndDate can be any of the following:

yyyy-MM-dd  
 yyyy-MM-ddTHH:mm:ss  
 yyyy-MM-ddTHH:mm:ss.SSSZ  
 yyyy-MM-ddTHH:mm:ss.mmmmmmmZ

### 5.6.2 GetRatingChanges Request

Field	Type	Nullable	Description
StartDate	DateTime	N	Date filter for the search, UTC time zone. StartDate and EndDate should be used to specify the past 24 hours, as this service should be called once every 24 hours. Format for both fields can be any of the following: yyyy-MM-dd

			yyyy-MM-ddTHH:mm:ss yyyy-MM-ddTHH:mm:ss.SSSZ yyyy-MM-ddTHH:mm:ss.mmmmmmmZ
EndDate	DateTime	N	Date filter for the search, UTC timezone.
MaxRows	Int32	N	Use this to specify how many rows/records per “page” (service will not allow more than 500 rows in the response). Example: This could be set to 50.
StartRowIndex	Int32	N	Use this to "move to next page". Example continued: 1) On first call, to get page 1, this would be set to 0. 2) Store would check to see if TotalRecordCount >=50; a) if <50, no further calls would be made b) else if >= 50, then the store knows there is another page to retrieve, so proceed to step 3. 3) On second call, to get page 2, StartRowIndex would be set to 50.

### 5.6.3 GetRatingChanges Response

Field	Type	Nullable	Description
TotalRecordCount	Int32	N	The store can use this to know how many “pages” of records exist (where page size is set by MaxRows, i.e., whatever page size the store would like to receive, as long as it’s less than 500.)
Result	Object	N	Indicates success or failure, and will include informative error messages for failures. When an exception is thrown, Result will be returned in the response without TotalRecordCount or CertList.
CertList	Object Array	N	Array of zero or more Cert objects, i.e. the search results.

#### Result Object

Field	Type	Nullable	Description
ResponseCode	String (enum)	N	Currently, the only values are either Success or Failure.
Messages	String	Y	For Success, will be empty. For Failure, will contain informative text.

### Cert Object

Field	Type	Nullable	Description
CertID	String (GUID)	N	IARC unique ID for the certificate
DateCreated	Date	N	Format: yyyy-MM-ddTHH:mm:ss.mmmmmmmZ
StoreProductID	String	Y	Store-assigned unique ID for the product.
RatingList	Object Array	Y	Array of zero or more Rating objects.

### Rating Object

Field	Type	Nullable	Description
RatingID	String (GUID)	N	IARC assigned globally unique ID for this rating decision by this rating authority. When a rating authority modifies a rating, that creates a new RatingID value. <ul style="list-style-type: none"> <li>Stores that host the questionnaire will ignore RatingID because ORC does not generate this field.</li> <li>Stores that use the IARC Hosted Configuration will receive RatingID in the PutCert service, and can use it in all interactions.</li> </ul>
RatingAuthorityID	Int32	N	IARC unique ID for the rating authority.
RatingAuthorityShortText	String	N	E.g. ESRB, PEGI, USK, ClassInd.
RatingLogicVersion	String	N	E.g. "5.1"
AgeRatingID	Int32	N	IARC unique ID for the age rating.
AgeRatingText	String	N	Display text for the age rating.
AgeRatingShortText	String	N	The short text representation of an age rating. Unique within a rating authority, no spaces, can be used as ID or for user display. Examples: ESRB: E, E10+, T, M, AO ClassInd: L, 10, 12, 14, 16, 18
NumericLevel	Int32	N	The ranking of the age rating within a rating authority's set of age ratings. E.g. ESRB has 5 age ratings; the lowest is E,

			which has NumericLevel 1, and the highest is AO, with NumericLevel 5.
DateCreated	Date	N	Format: yyyy-MM-ddTHH:mm:ss.mmmmmmmZ
DescriptorList	Object Array	N	Array of 0 or more Descriptor objects, for this certificate under this rating authority.
InteractiveElementList	Object Array	N	Array of 0 or more InteractiveElement objects.

### Descriptor Object

Field	Type	Nullable	Description
DescriptorID	Int32	N	IARC unique ID for the descriptor.
DescriptorText	String	N	The value will be a meaningful string ID to be used for looking up localized display text (we call this a messageID within IARC). That display text will be a content descriptor assigned by a rating authority, e.g. ESRB might assign “Blood and Gore” or “Crude Humor”.
DisplayOrder	Int32	N	When multiple descriptors are present for a given rating authority, they are in a specific order according to the rules of that rating authority, and should be shown in the UI in that order.

### InteractiveElement Object

Field	Type	Nullable	Description
InteractiveElementID	Int32	N	IARC unique ID for the interactive element.
InteractiveElementText	String	N	Value will be an ID to look up display text. IDs: <ul style="list-style-type: none"> <li>• IE_UsersInteract</li> <li>• IE_SharesInfo</li> <li>• IE_SharesLocation</li> <li>• IE_DigitalPurchases</li> <li>• IE_UnrestrictedInternet</li> </ul> Display text in English: <ul style="list-style-type: none"> <li>• Users Interact</li> <li>• Shares Info</li> <li>• Shares Location</li> <li>• Digital Purchases</li> </ul>

			<ul style="list-style-type: none"> <li>Unrestricted Internet</li> </ul>
--	--	--	---

## 5.7 UpdateCerts

Used In Configurations: IARC Hosted and Store Hosted

### 5.7.1 Business Logic

This service is used to update IARC with a change to store-specific attributes tied to a cert. These include:

- whether or not the product is live on the store (i.e. available for rating authorities to review it)
- IARC contact email address for the developer
- localized display names for the product
- platforms the product is available on
- countries the product is available in
- whether the store deems the cert invalid because of user error, i.e. if the developer answered the questionnaire incorrectly

### 5.7.2 UpdateCerts Request

Field	Type	Nullable	Description
UpdateList	Object Array	N	Array of 1 to 100 Update objects.

#### Update Object

Field	Type	Nullable	Description
CertID	String (GUID)	N	Globally unique ID for the certificate
LiveOnStore	Bool	Y	<p>Indicates whether the product is currently live on the storefront, i.e. available for the rating authorities to review the product page that displays the IARC cert.</p> <p>This field should only be present if either</p> <ul style="list-style-type: none"> <li>• the product has changed from not live to live (LiveOnStore=true)</li> <li>• the product has changed from live to not live (LiveOnStore=false)</li> </ul> <p>If neither of those events have occurred, this field should not be supplied.</p> <p>(This replaces the old Action=Publish and Action=UnPublish.)</p>
CertInvalid	Bool	Y	This lets the store inform IARC that the store has decided the cert is invalid, e.g. if the store feels the

			<p>developer answered the questionnaire incorrectly. IARC may choose to review a cert that a store has invalidated. Note that the cert may still be in use on other stores.</p> <p>(This replaces the old Action=InvalidateCert.)</p>
DeveloperEmail	String	Y	The developer's contact address for IARC communications.
DeveloperID	String	Y	<p>Store-assigned unique ID for the developer. This will only be included when the store wants to update the developer's email address.</p> <p>If you provide this field, IARC will update the email address for all certs tied to this StoreDeveloperID; if you omit this field for the same operation, IARC will update only the email address on this CertID.</p>
ProductID	Int32	Y	The store's internal unique ID for the product. See Business Logic section above for usage details.
CountryCodes	String Array	Y	A list of country codes that specify where this product is available. See CreateCerts request for full description.
Platforms	String Array	Y	A list of the platforms for which this product is available. See CreateCerts request for full description.
LocalizedTitles	Object Array	Y	Array of LocalizedTitle objects. See CreateCerts request for full description.

#### LocalizedTitle Object

Field	Type	Nullable	Description
Title	String (UTF-8)	N	Localized UTF-8 display text.
LanguageID	String	N	<p>Same as specified elsewhere in this doc:</p> <ul style="list-style-type: none"> <li>- A two-letter ISO 639-1 language designator</li> <li>- Followed by a two-letter capitalized ISO 3166-1 alpha-2 region designator</li> </ul>

#### 5.7.3 UpdateCerts Response

Field	Type	Nullable	Description
ResultList	Object	N	Array of Result objects.

	Array		For every item in UpdateList in the request, there will be a corresponding Result in this list.
--	-------	--	---

### Result Object

Field	Type	Nullable	Description
CertID	String (GUID)	N	Globally unique ID for the certificate.
Action	String	N	The action that was requested.
ResultCode	String	N	Value is either Success or Failure.
ErrorID	String (GUID)	Y	When ResultCode=Failure, this will be a globally unique ID that the IARC team can use for locating the event in the system logs.
ErrorMessage	String	Y	When ResultCode=Failure, this will contain human readable descriptions of the error condition.

## 6 Offline Ratings Calculator

**Used In Configurations: Store Hosted**

### 6.1 Overview

The Offline Ratings Calculator (ORC) runs on the store's servers and allows the store to locally generate ratings, content descriptors, and interactive elements based on IARC questionnaire responses. ORC is available as either a .Net DLL or Java JAR file.

To use the .Net ORC, you will add a reference to IARC.ORB; instantiate the OfflineRatingCalculator class; and call the CalculateRatings method on that object, passing the developer-chosen ResponseInstanceIDs as IEnumerable<int>. CalculateRatings returns an ORCOutput object, which is described below. Implementation details for Java are similar and available upon request.

### 6.2 ORC Input

The store hands to ORC a comma separated list containing all the ResponseInstanceIDs that were selected by the developer. Each rating authority defines its own rules that will be triggered by certain response IDs. These rules result in ORC returning for each rating authority one age rating, and zero or more content descriptors. Separate from the rating authorities, IARC rules will also output zero or more interactive elements.

### 6.3 ORC Output

The fields of the ORCOutput object are described below, they are a subset of the JSON object returned by IARC web services in that ORC does not output localized display text.

#### ORCOutput Object

Field	Type	Description
-------	------	-------------

RatingLogicVersion	String	<p>The version of the rating logic used to generate this rating data. This will only change when the rating authorities need to make a change to rating logic. This version will always be of the form <i>major.minor</i>, e.g. 5.2.</p> <p>Major changes will increment the major version (first number, e.g. 5.2 to 6.0), which historically has not happened more than once a year.</p> <p>Minor changes, such as changing the rating logic for one question, will increment the minor version (second number, e.g. 6.0 to 6.1).</p>
ORCVersion	String	<p>The version of the Offline Ratings Calculator codebase itself, as distinct from the rating logic. This version will always be of the form <i>major.minor.patch</i>, e.g. 6.1.0.</p> <p>A change to rating logic requires a change to this codebase, so this major and minor version will always match the rating logic version. But when there is a need for a codebase change without a need for a rating logic change, we will increment the patch version (third number, e.g. 6.1.0 to 6.1.1).</p> <p>Thus, a rating logic change always pushes up both the rating logic version and the ORC version, but an ORC-only change pushes up only the ORC version.</p>
RatingList	Array of ORCRating	<p>The set of ratings data generated by ORC from the questionnaire responses. Array of zero or more ORCRating objects.</p>

### ORCRating Object

Field	Type	Description
RatingAuthorityID	Int32	IARC unique ID for the rating authority.
RatingAuthorityShortText	String	Short version of the rating authority name, e.g. ESRB, PEGI, USK, ClassInd. This can be used for display text.
AgeRatingID	Int32	IARC unique ID for the age rating provided by this authority.
AgeRatingShortText	String	The short text representation of an age rating. Unique within a rating authority, no spaces, can be used as ID or for user display.

		Examples: ESRB: E, E10+, T, M, AO ClassInd: L, 10, 12, 14, 16, 18
NumericLevel	Int32	The ranking of the age rating within a rating authority's set of age ratings. E.g. ESRB has 5 age ratings; the lowest is E, which has NumericLevel 1, and the highest is AO, with NumericLevel 5.
DescriptorList	Array of ORCDescriptor	Array of 0 or more ORCDescriptor objects, for this certificate under this rating authority.
InteractiveElementList	Array of ORCInteractiveElement	Array of 0 or more ORCInteractiveElement objects.

#### ORCDescriptor Object

Field	Type	Description
DescriptorID	Int32	IARC unique ID for the descriptor.
DescriptorText	String	The value will be a meaningful string ID to be used for looking up localized display text (we call this a messageID within IARC). That display text will be a content descriptor assigned by a rating authority, e.g. ESRB might assign "Blood and Gore" or "Crude Humor".
DisplayOrder	Int32	When multiple descriptors are present for a given rating authority, they are in a specific order according to the rules of that rating authority, and should be shown in the UI in that order.

#### ORCInteractiveElement Object

Field	Type	Description
InteractiveElementID	Int32	IARC unique ID for the interactive element.
InteractiveElementText	String	Value will be an ID to look up display text. IDs: <ul style="list-style-type: none"> <li>• IE_UsersInteract</li> <li>• IE_SharesInfo</li> <li>• IE_SharesLocation</li> <li>• IE_DigitalPurchases</li> <li>• IE_UnrestrictedInternet</li> </ul>

		<p>Display text in English:</p> <ul style="list-style-type: none"> <li>• Users Interact</li> <li>• Shares Info</li> <li>• Shares Location</li> <li>• Digital Purchases</li> <li>• Unrestricted Internet</li> </ul>
--	--	--

## 6.4 Versioning

Each questionnaire is tied to a RatingLogicVersion. When IARC creates a new RatingLogicVersion, IARC will provide to the store a corresponding version of ORC to use. Thus the store will have one copy of ORC for each RatingLogicVersion.

- When a developer has completed a questionnaire, the store will check their questionnaire tables for the RatingLogicVersion used by that questionnaire.
- The store will pass the list of trigger responseIDs to that version of ORC
- ORC output will include the RatingLogicVersion running on that instance of ORC, so that the store's system can verify the version is correct.

## 6.5 Handling Errors & exceptions

Data validation exists throughout ORC, and exceptions will be thrown in error scenarios. Currently ORC expects the responseIDs to be valid and unique. If it encounters a trigger that it does not recognize, it stops the evaluation process, throws an exception, and logs this error to the console. Such exceptions should not happen as long as the store invokes the version of ORC that has the same RatingLogicVersion as the questionnaire presented to the developer.

## 7 Reference Data

The IARC team will supply to the store a set of pipe delimited flat files for the following IARC database tables. All fields are non-nullable.

### 7.1 Message Table

All IARC display texts are in this localized table. PK is comprised of messageID and languageID together.

Fields:

- **messageID** (varchar(40)): IARC unique identifier for this display text
- **languageID** (varchar(8)): ISO language codes as described in the Localization sections above
- **text** (nvarchar(max)): the translated text to be displayed to the user (UTF-8)

### 7.2 RatingBoard Table

Contains all IARC rating boards.

Fields:

- **ratingBoardID** (int): PK, IARC unique ID for the rating board
- **name** (varchar(50)): short name, e.g. PEGI, ClassInd
- **longName** (nvarchar(50)): full length name, e.g. “Pan European Game Information”, “Classificação Indicativa”

### 7.3 AgeRating Table

Contains age ratings for all rating boards.

Fields:

- **ageRatingID** (int): PK, IARC internal ID for the age rating
- **ratingBoardID** (int): FK to RatingBoard table
- **shortName** (varchar(10)): short text e.g. “E10” (the store may or may not have use for this field)
- **name** (varchar(40)): full length non-localized text that the store will display, e.g. “Everyone 10+”
- **numericLevel**: the age ratings for the rating board represented as a zero-based index, e.g. for ESRB 0=E, 1=E10, 2=Teen, 3=Mature, 4=Adults Only
- **ratingRefused** (bool): this table is essentially a table of rating board *decisions*, and records in this table that are not actually an age rating will have this field set to true. Those are: USK Refused Classification, ACB Refused Classification, and ACB Not Applicable. RC applies to products that the rating board decided not to rate based on developer responses; ACB only rates games, so ACB will issue an N/A for all products that are not a game.

Note that the rating board named Generic does not have its own sets of age ratings or descriptors, it uses PEGI’s.

### 7.4 Descriptor Table

Contains all descriptors for all rating boards.

Fields:

- **descriptorID** (int): PK, IARC internal ID for the descriptor
- **ratingBoardID** (int): FK to RatingBoard table
- **name** (varchar(40)): the value in this field is a messageID from the Message table. So to show the descriptor name to the user, look up the *text* for this *messageID* in the Message table, along with the desired *languageID*.
- **shortName** (varchar(10)): internal IARC ID that is unique for this descriptor *within this rating board*. The store may or may not have a use for this field.

### 7.5 InteractiveElement table

Contains all interactive elements. Interactive elements are commonly referred to as IEs. This one set of IEs is shared by all rating boards; conceptually, an IE is generated by IARC, not by any individual rating board.

Fields:

- **interactiveElementID** (int): PK, IARC internal ID for the IE
- **name** (varchar(40)): same as with Descriptor table above: this fields contains a messageID value to be looked up in the Message table

- **shortName** (varchar(5)): internal IARC ID that is unique for this IE. The store may or may not have a use for this field.

## 8 Guide For Display Of Rating Information On Digital Storefronts\*

\*Includes mobile (e.g., phone and tablet), console, and online storefronts where consumers download or access games or apps.

- **Rating Icons** indicate the age-appropriateness of a game/app and differ depending on the region.
- **Content Descriptors** indicate the reason(s) (e.g., Violence, Sexual Content, Language) that a game/app was assigned a particular age rating and may differ depending on the region.
- **Interactive Elements** indicate other interactive aspects of a game/app (e.g., if it displays the user's location to others, allows the user to create or exchange content or make in-app purchases) and are the same across regions.

### 8.1 Optimal Rating Display

The assigned rating icon, content descriptor(s) and interactive element(s) are all displayed above the fold on the product detail page.



### 8.2 Specifications

The rating icon **must** be displayed on either the product detail or purchase page, and link to a ratings definition page from its respective rating authority.

**Content Descriptors and Interactive Elements:**

- Should be displayed beside or underneath the rating icon in text that is large and clear enough to be legible to most consumers.
- The first letter of each word should be capitalized and the rest of the letters should be in lower case letters.
- Should not be mixed together. The content descriptors should be displayed first followed by the interactive elements, separated by a line or space.

### Rating Information Should Always:

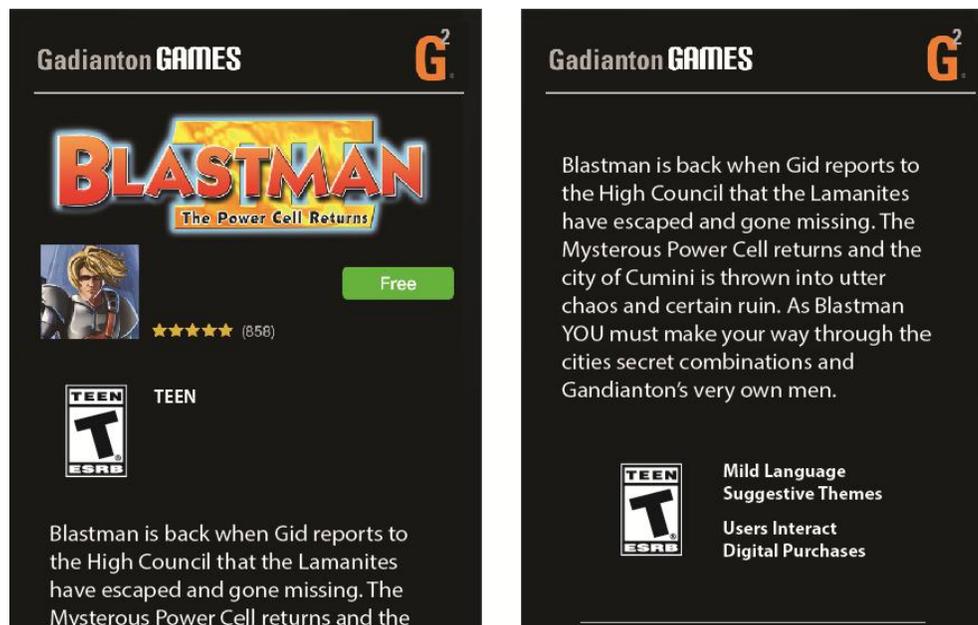
- Appear in the same location across a given device or storefront where it will be accurately associated with the rated game/app.
- Be legibly displayed where it will be visible to the user prior to downloading or purchasing the game/app (e.g., on the product detail page) in the rating authority’s respective region.
- At some point be displayed together. If all of the rating information cannot be displayed together above the fold, see Alternate Rating Display below.

### 8.3 Alternate Rating Display

The assigned rating icon is displayed above the fold with text of rating category next to the icon

AND

complete assigned rating information (i.e., rating icon, content descriptors and interactive elements) are all displayed together below the fold.



Rating icons and rating definition pages for all participating IARC rating authorities are available in the IARC Download Library: [globalratings.com/downloads](http://globalratings.com/downloads)



Rating components are trademarks of their respective rating authorities and cannot be altered in any way.