IEC 62769-6-200

Edition 1.0 2023-04

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Field Device Integration (FDI®) – Part 6-200: Technology Mapping – HTML5

Intégration des appareils de terrain (FDI®) – Partie 6-200: Mapping de technologies – HTML5

IEC 62769-6-200:2023-04(en-fr)

AVA Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

THIS PUBLICATION IS COPYRIGHT PROTECTED
Copyright © 2023 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Secretariat
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

About the IEC
The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications
The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigendum or an amendment might have been published.

IEC publications search - webstore.iec.ch/advsearchform
The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee, …). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

IEC Customer Service Centre - webstore.iec.ch/csc
If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch
Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

Electropedia - www.electropedia.org
The world's leading online dictionary on electrotechnology, containing more than 22 300 terminological entries in English and French, with equivalent terms in 19 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

A propos de l'IEC
La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC
Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Recherche de publications IEC - webstore.iec.ch/advsearchform
La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études, …). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished
Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et une fois par mois par email.

Service Clients - webstore.iec.ch/csc
Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch
Découvrez notre puissant moteur de recherche et consultez gratuitement tous les aperçus des publications. Avec un abonnement, vous aurez toujours accès à un contenu à jour adapté à vos besoins.

Electropedia - www.electropedia.org
Le premier dictionnaire d'électrotechnologie en ligne au monde, avec plus de 22 300 articles terminologiques en anglais et en français, ainsi que les termes équivalents dans 19 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

IEC 62769-6-200

Edition 1.0 2023-04

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Field Device Integration (FDI®) – Part 6-200: Technology Mapping – HTML5

Intégration des appareils de terrain (FDI®) – Partie 6-200: Mapping de technologies – HTML5

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

ICS 25.040.40; 35.100.05

ISBN 978-2-8322-6820-9

Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

® Registered trademark of the International Electrotechnical Commission
Marque déposée de la Commission Electrotechnique Internationale

AVA Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

CONTENTS

FOREWORD 4
1 Scope 6
2 Normative references 6
3 Terms, definitions, abbreviated terms, acronyms and conventions 6
3.1 Terms and definitions 6
3.2 Abbreviated terms and acronyms 7
3.3 Conventions 7
4 Technical concepts 7
4.1 General 7
4.1.1 Overview 7
4.1.2 FDI® Type Library 7
4.2 UIP representation 9
4.2.1 Self containment 9
4.2.2 External libraries and supplementary files 10
4.3 UIP compatibility rules 10
4.3.1 Overview 10
4.3.2 FDI® Type Library Compatibility Strategy 10
4.3.3 Runtime compatibility strategy 10
4.3.4 UIP executable compatibility rules 11
4.4 UIP Deployment 11
4.5 UIP Life-cycle 11
4.5.1 General 11
4.5.2 UIP activation steps 12
4.5.3 UIP deactivation 13
4.5.4 General rules for UIP activation and deactivation 13
4.6 Interaction between an FDI® Client and a UIP 14
4.6.1 Handling of standard UI elements 14
4.6.2 Non-blocking service execution 14
4.6.3 Threading 16
4.6.4 Timeout 17
4.6.5 Exception handling 17
4.6.6 Type safe interfaces 18
4.6.7 Globalization and localization 18
4.7 Security 19
4.7.1 General 19
4.7.2 Access permissions 19
5 Interface definition 20
Bibliography 25

Figure 1 – FDI® Type Library structure 8
Figure 2 – Relationship between FDI® Type Library, host specific and UIP specific implementation of the interfaces 8
Figure 3 – Examples of UIP supplementary data files include javascript files, css files, and images 9
Figure 4 – Compatibility strategy to host UIPs built for different Runtime Versions 11

AVA Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

Figure 5 – UIP Life-cycle 12
Figure 6 – UIP activation process 13
Figure 7 – Asynchronous service execution example (FDI® Client) 15
Figure 8 – Cancelled asynchronous service execution example (FDI® Client) 15
Figure 9 – Asynchronous service execution example (UIP) 16
Figure 10 – Simplified code sample to illustrate the described function implementation 16
Figure 11 – Component interaction 18

Table 1 – Base Property Services 20
Table 2 – Device Model Services 21
Table 3 – Access Control Services 21
Table 4 – Direct Access Services 21
Table 5 – Hosting Services 22
Table 6 – UIP Services 23
Table 7 – Base Data Types 23
Table 8 – Special Types 24
Table 9 – Parameter Types 24

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FIELD DEVICE INTEGRATION (FDI®) –

Part 6-200: Technology Mapping – HTML5

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

IEC c0 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation. It is an International Standard.

The text of this International Standard is based on the following documents:

Draft	Report on voting
65E/870/CDV	65E/926/RVC

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English.

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/publications.

A list of all parts in the IEC 62769 series, published under the general title *Field device integration (FDI®)*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under webstore.iec.ch in the data related to the specific document. At this date, the document will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The "colour inside" logo on the cover page of this document indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

FIELD DEVICE INTEGRATION (FDI®) –

Part 6-200: Technology Mapping – HTML5

1 Scope

This part of IEC 62769 specifies the technology mapping for the concepts described in the Field Device Integration (FDI®[1]) standard. The technology mapping focuses on implementation regarding the components FDI® Client and User Interface Plug-in (UIP) for the Runtime HTML5.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

FCG TS10099, *Field Device Integration (FDI®) – Technology Management*

IEC 62769-1:2021, *Field device integration (FDI®) – Part 1: Overview*

IEC 62769-2:2021, *Field device integration (FDI®) – Part 2: Client*

IEC 62769-4, *Field device integration (FDI®) – Part 4: FDI® Packages*

IEC 62769-6-100, *Field Device Integration (FDI®) – Part 6-100: Technology Mapping – .NET*

W3C HTML5.0, *W3C Recommendation HTML5 – A vocabulary and associated APIs for HTML and XHTML*

3 Terms, definitions, abbreviated terms, acronyms and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62769-1, IEC 62769-6-100, as well as the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http:www.iso.org/obp

[1] FDI is a registered trademark of the non-profit organization Fieldbus Foundation, Inc. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the trademark holder or any of its products. Compliance does not require use of the trade name. Use of the trade name requires permission of the trade name holder.

AVA Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

3.1.1
HTML5 runtime
functional component of the FDI® Client, which executes the HTML5-based UIP. In principle, it provides the functionality of a web browser (HTML rendering, JavaScript execution engine)

3.1.2
FDI® Host Type Library
host specific implementation of FDI® Type Library which consists of host.js and fdi.js. FDI® Host Type Library is provided by the host vendors and will be deployed together with the UIP on UIP startup

3.2 Abbreviated terms and acronyms

For the purposes of this document, the abbreviated terms and acronyms given in IEC 62769-1 as well as the following apply.

UML Unified Modeling Language

3.3 Conventions

Figures in this document use the graphical symbols according to ISO/IEC 19505-1 (UML 2.0).

4 Technical concepts

4.1 General

4.1.1 Overview

In 4.1.2, this document describes the technology base for UIP implementation based on HTML5, the software environment including the related implementation rules. Clause 4 follows a life-cycle (use case) oriented approach.

Subclause 4.3.4 describes the copy deployment procedures and related implementation rules for the UIP and the FDI® Client. UIP instantiation and termination is described in 4.5. Subclause 4.6 defines the rules about interaction between the FDI® Client and the UIP. Security related definitions are written in 4.7. The service interface definitions for the FDI® Client and the UIP are found in Clause 5.

4.1.2 FDI® Type Library

The Device Access Services, the Hosting Services, and the UIP Services are modelled as TypeScript interfaces passing TypeScript data type arguments. These interfaces and data types are used for the data exchange and interaction between the UIP and the FDI® Client. For runtime error handling purposes during interface method calls TypeScript error objects are defined.

The FDI® TypeScript interfaces, data types, and exception classes are defined in a single FDI® Type Library. The file name of the interface definition shall be ‘fdi.ts’.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

Figure 1 shows the FDI® Type Library structure.

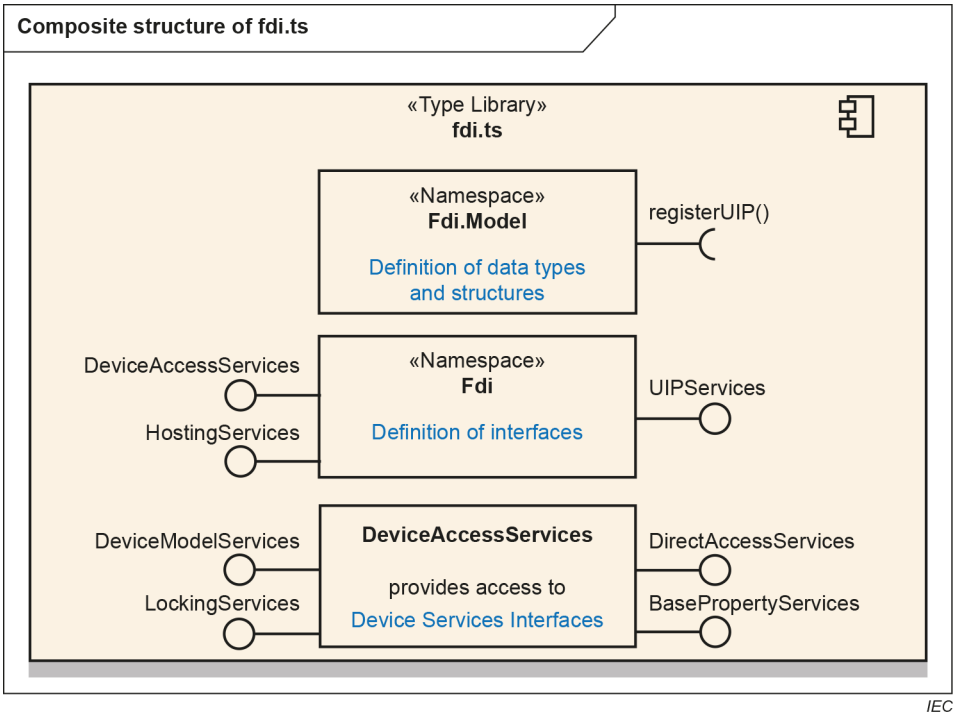


Figure 1 – FDI® Type Library structure

NOTE 1 The composite structure diagram shows only the core interfaces that implement the interfaces defined in IEC 62769-2.

The interfaces shall be implemented by the FDI® Client respectively the UIP using TypeScript. The TypeScript implementation files are transpiled to JavaScript using the feature set specified in the edition of ECMA 262, which corresponds to the RuntimeId of the UIP variant. The edition of ECMA 262 is unambiguously specified in the FCG TS10099 by the RuntimId of the UIP variant.

The result of the transpilation of FDI®.ts is FDI®.js, which contains the function prototypes. The transpiled FDI® Type Library, i.e. FDI®.js, and the host specific implementation of the interfaces, i.e. host.js, is called FDI® Host Type Library. The overall structure of FDI® Type Library together with host and UIP specific implementations of the interfaces is shown in Figure 2.

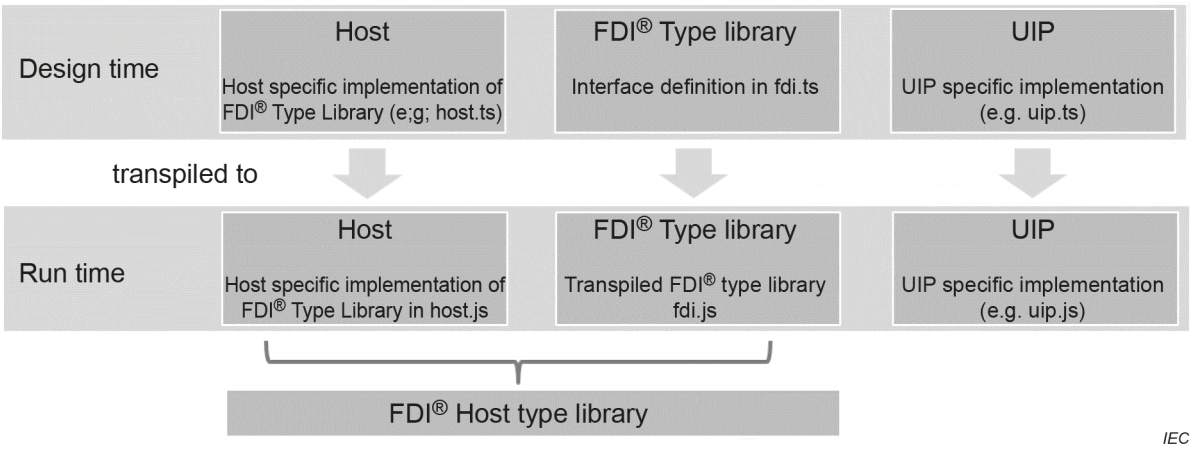


Figure 2 – Relationship between FDI® Type Library, host specific and UIP specific implementation of the interfaces

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

NOTE 2 The FDI® Type Library is specified by this specification and can be obtained from FieldcommGroup (see www.fieldcommgroup.org). Host specific implementation will be provided by host vendors and the UIP specific implementation will be part of the specific UIP variant.

The FDI® Type Library fdi.ts respectively the fdi.js shall be versioned as per IEC 62769-1:2021, 8.1. The FDI® Type Library is part of the FDI® Core Technology as per IEC 62769-1:2021, 8.3.2.1. and therefore directly influences the FDI® Technology Version. All Compatible changes of the fdi.ts respectively fdi.js lead to an increase of the minor portion of the FDI® Technology Version. Incompatible changes lead to an increase of the major portion of the FDI® Technology Version (see IEC 62769-1:2021, 8.3.2.2).

The FDI® Type Library shall be installed as part of every FDI® Client installation. User Interface Plug-Ins (UIP) and the FDI® Client Application shall use this instance of the fdi.js. UIPs shall not carry or deploy the FDI® Type Library. The FDI® Client is responsible to provide means to allow updates of this type library over time.

The transpiled host specific implementation of the FDI® Type Library, i.e. host.js, and the FDI® Type Library itself (fdi.js) shall be provided by the FDI® Client during the load process of the UIP. Both files shall be located in the subfolder ‘./scripts’ of the UIP installation folder. The respective files of the FDI® Host Type Library fdi.js and host.js shall be referenced by the UIP explicitly. For example, this can be done with the statements <script src="./scripts/ fdi.js" type="module"/> and <script src="./scripts/host.js" type="module"/> within the <header>-statement of the HTML file named according to the <StartElementName>-Attribute of the HTML5 UIP Package.

4.2 UIP representation

4.2.1 Self containment

The UIP Variant can contain either a single or multiple HTML files and their related supplementary files (for example resource files). The main entry HTML file is located directly in the UIP installation directory. Its name is given by the `<StartElementName>`-Attribute of the HTML5 UIP Package.

Besides the FDI® Host Type Library, i.e. FDI®.js and host.js, the UIP Variant shall be self-contained. All supplementary files required by the UIP are stored under (a) subfolder(s) of the installation directory. The FDI® Host Type Library is provided by the FDI® Client.

Figure 3 shows an example of a UIP with subfolders for scripts, Cascading Style Sheets and images.

```
async function browseParmeterList() {

    var node = new UIPTypes.NodeSpecifier("parameterSet", false);                    // create NodeSpecifier to be browsed
    var cancelToken = new UIPTypes.CancelToken(guid());                              // create cancel token to enable cancelling of call
    tokenDirectory[cancelToken.id] = cancelToken;

    let fdiDMS = deviceModelServices;                                                // interface to host specific implementation of device model services

    await fdiDMS.browse(node,cancelToken).then(                                      // async call to browse function
        response => { BrowseResponse(response, "Browse"); },                         // response evaluation, if call returns successful
        error => {                                                                   // response evaluation, if call returns wit failure
            if(error == Fdi.Model.StatusCode.Bad_RequestCancelled){                  // call was cancelled
                log("async call to Browse with ID:" + cancelToken.id + " cancelled");
                delete tokenDirectory[cancelToken.id];
            }
            else{                                                                    // other error inside browse function call
                log("async call to Browse with ID:" + cancelToken.id + " failed");
                delete tokenDirectory[cancelToken.id];
            }
        }
    ).catch(                                                                         // call to browse service failed
        exception => { log("async call to Browse with ID:" + cancelToken.id + " catched");
                delete tokenDirectory[cancelToken.id];
        }
    );
};
```

IEC

Figure 3 – Examples of UIP supplementary data files include javascript files, css files, and images

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

4.2.2 External libraries and supplementary files

The UIP can depend on external libraries (third party libraries) and other components, for example, specific user control libraries, icons, or images. FDI® Clients dynamically load and execute the UIP in the HTML5 runtime. To support this usage, as well as the requirement to prevent possible problems of conflicting libraries, the following rules are specified for external libraries and supplementary files.

External libraries and supplementary files shall:

- be contained within the UIP variant, except of the FDI® Host Type LIbrary (FDI®.js and host.js), which are provided by the FDI® Client;
- not require any installation procedure other than copying into the UIP installation directory or its subfolders;
- adhere to the access restrictions described in 4.7.2;
- be compatible with the platforms and runtimes the UIP Variant is built for.

The FDI® Client loads the HTML5 UIP by navigating to the main entry file given by the `<StartElementName>`-Attribute of the HTML5 UIP Package. Referenced external libraries and supplementary files have to be loaded explicitly by the UIP itself (e.g. stated as <script src="./scripts/myExternalLibrary.js" />). All references to external libraries and supplementary files within the UIP shall be specified as relative URLs.

4.3 UIP compatibility rules

4.3.1 Overview

The compatibility rules for different versions of the UIP variant are specified in IEC 62769-4.

An FDI® Client shall provide the versions of the FDI® Host Type Library (fdi.js andhost.js) and the HTML5 runtime supporting the feature set, which correspond to the RuntimeId of the UIP variant.

4.3.2 FDI® Type Library Compatibility Strategy

The UIP shall check for the FDI® Technology Version of the FDI® Client using the HostingService GetClientTechnologyVersion. The following rules ensure the compatibility of the UIP with the FDI® Host Type Library:

- If the UIP variant was built for a completely compatible FDI® Technology Version (same major and minor version), the UIP can be directly started in the FDI® Client.
- If the minor version of the FDI® Technology of the FDI® Client is lower than the minor version of FDI® Technology the UIP has been built for, the UIP may only use the interface methods defined in the FDI® Technology Version supported by the FDI® Client.
- If the UIP variant to be started was built for an incompatible FDI® Technology Version (different major versions), the FDI® Client shall start the UIP using the matching FDI® Host Type Library and Runtime.

4.3.3 Runtime compatibility strategy

In different runtime versions, UIPs may use a different, potentially incompatible feature set. The feature set, which shall be supported by any runtime version identified by a specific RuntimeId is specified in the FCG TS10099.

FDI® Clients supporting a specific RuntimeId shall support at least the features listed in the FCG TS10099 for the individual RuntimeId.

Documents
Document Sharing Hub

Full document details: IEC 62769-6-200-2023

UIP variants built for a specific RuntimeId can rely on the features specified in the FCG TS10099 for this RuntimeId. If a UIP uses features, which are not listed in the standard feature set of the corresponding RuntimeId, the UIP shall check for the availability of the feature in the respective FDI® Client and implement a fallback strategy as for example an information message to the user.

If an FDI® Client detects that a UIP variant is built for a RuntimeId requiring another feature set from the runtime, the FDI® Client can use the appropriate runtime, e.g. the HTML5+ runtime, to execute the UIP (see Figure 4).

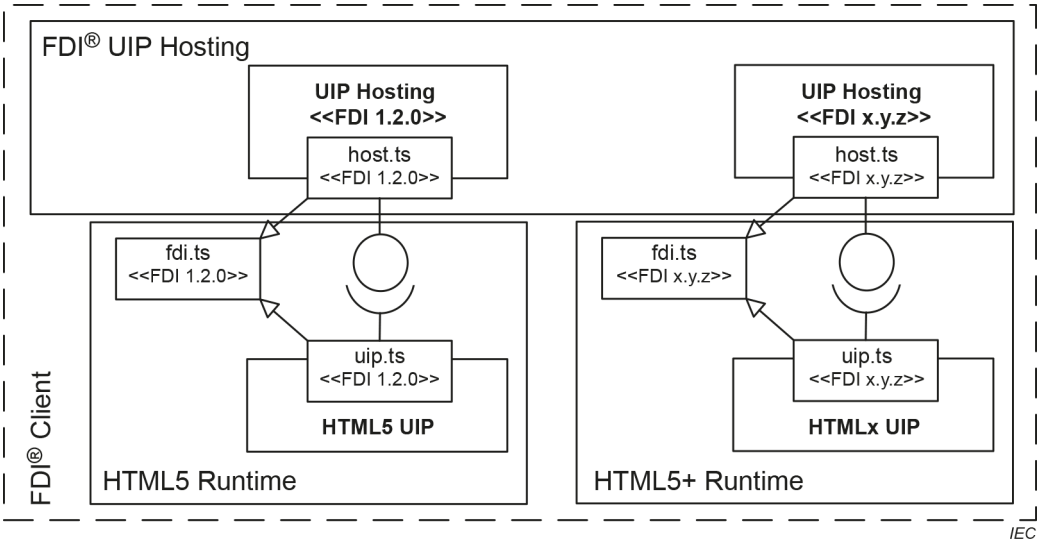


Figure 4 – Compatibility strategy to host UIPs built for different Runtime Versions

4.3.4 UIP executable compatibility rules

A HTML5 based User Interface Plugin is not compiled to machine code. Therefore, the element "CpuInformation" in the catalog.xml file, which holds the compilation target platform information, shall not be used for HTML5 based User Interface Plugins.

4.4 UIP Deployment

The general UIP installation rules are outlined in IEC 62769-2. If the UIP is stored on the local file system, the FDI® Client and the FDI® Server shall ensure the integrity of the UIP.

The FDI® Client implementation ensures that UIP deployment works independently from current user credentials. (See the note below.)

NOTE Certain operating system managed folders require specific access rights, for example, modifications in folder "Program Files" require "Administrator" rights. The Windows operating system provides several means to allow an application running with restricted user rights, to execute actions with administrator privileges transparent to the user, for example, special restriction handling for identified directories, services with administration rights, executables that are configured to automatically run with administration rights. The alternative is to copy UIP into folders writeable for "normal" users.

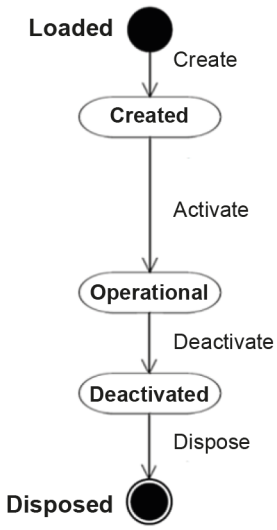
4.5 UIP Life-cycle

4.5.1 General

The UIP state machine, outlined in IEC 62769-2, is composed of the Loaded, Created, Operational, Deactivated and Disposed states. The mechanisms affecting state changes are described in 4.5.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

After UIP deployment (see 4.3.4), the FDI® Client loads the HTML5 runtime dynamically into the memory and executes the related logic of the UIP by calling the corresponding FDI® specified interface functions. Figure 5 gives an overview on the UIP life-cycle.



Transition step	Source step	Destination step	Description
Create	Loaded	Created	The FDI® Client loads its HTML5 runtime, creates a browsing context for the UIP and navigates this browsing context to the main HTML entry page given by the <StartElementName>-Attribute of the HTML5 UIP Package. All resources referenced in the main HTML entry page shall be loaded as specified in W3C HTML5.
Activate	Created	Operational	The HTML5 Runtime sends the Javascript event "onload" to the UIP. The UIP shall bind its activation method to this event. Within this method it implements its startup and initialization sequence, which includes the registration of UIPServices interface implementation on the FDI® Client by calling Fdi.Model.registerUIP(). Finally, the FDI® Client calls UIPServices.activate() and the UIP reaches the state "Operational"
Deactivate	Operational	Deactivated	The FDI® Client shall call the UIPServices.deactivate() method by using the registered UIPServices interface. The UIP shall finish its operations and store data. The UIP shall bring itself into a state where it can be safely released.
Dispose	Deactivated	Disposed	The FDI® Client shall delete all references to the UIP instance.

IEC

Figure 5 – UIP Life-cycle

Subclauses 4.5.2 and 4.5.3 specify the rules about how the FDI® Client shall activate and deactivate the UIP.

4.5.2 UIP activation steps

4.5.2.1 Load

The FDI® Client shall load its HTML5 runtime. Within the HTML5 runtime the creation of the UIP shall be done by creating a browsing context for the UIP and navigating this browsing context to the main HTML entry page given by the `<StartElementName>`-Attribute of the HTML5 UIP Package. The `<StartElementName>`-Attribute is contained in the file uipcatalog.xml of the UIP Package.

4.5.2.2 Create

The UIP is created within the HTML5 runtime. All resources referenced in the main HTML entry page shall be loaded as specified in W3C HTML5.0.

When the UIP is created successfully, the HTML5 runtime shall send the Javascript event "onload" to the UIP.

4.5.2.3 Activate

The UIP shall bind its startup method to the Javascript event "onload". Within this method, the UIP implements its startup and initialization sequence.

Within the startup method the UIP shall call the method `Fdi.Model.registerUIP()` to enable the FDI® Client to call the methods on the UIP as specified in IEC 62769-2:2021, 6.1 UIP Services.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

After registration of the UIP Services interface implementation of the specific UIP, the FDI® Client shall call the method `Fdi.UIPSerice.setSystemLabel()` to specify a human identifier of the UIP instance in the context of the FDI® Client. To finish UIP activation, the FDI® Client calls `Fdi.UIPSerice.activate()`. The invocation of `Fdi.UIPSerice.activate()` includes information on localization (RegionInfo, CultureInfo) and the calling context including the FDI® client specific implementation of the interfaces DeviceAccessServices and HostingServices. With successful completion of `Fdi.UIPSerice.activate()` the activation process of the UIP is finished and the UIP is operational.

The whole activation process is shown in Figure 6.

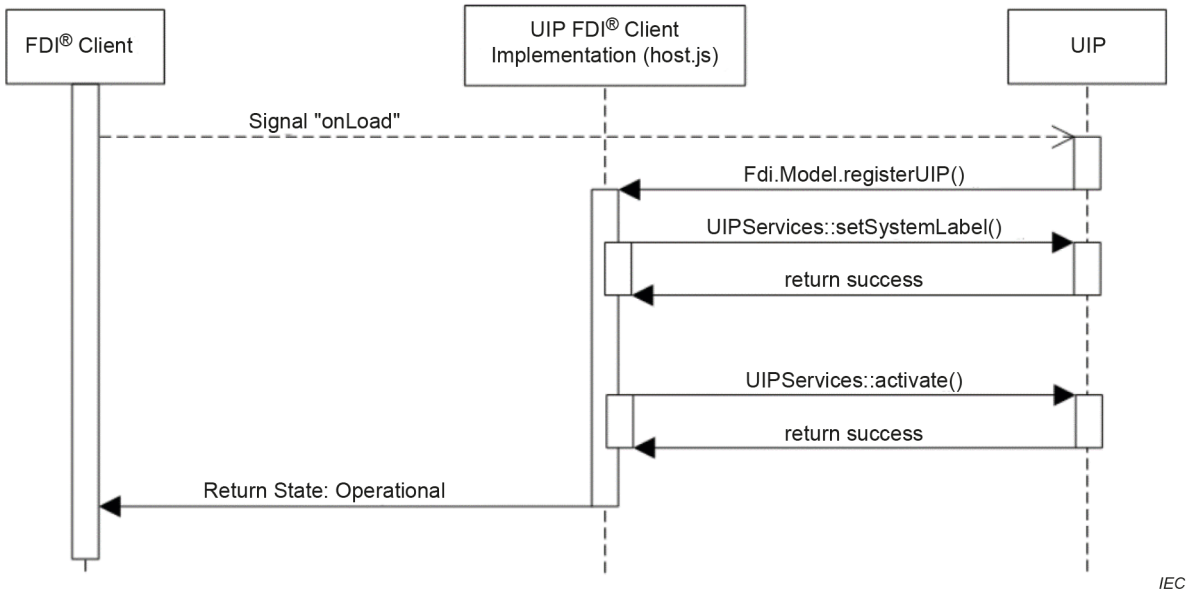


Figure 6 – UIP activation process

4.5.3 UIP deactivation

For UIP deactivation the FDI® Client shall call the method `Fdi.UIPService.deactivate()` using the implementation of the UIP Services interface registered during the activation process.

Afterwards the FDI® Client shall delete all references to the UIP instance.

4.5.4 General rules for UIP activation and deactivation

The activation logic shall be limited to instantiate the object in terms of the internal data structure and load of all referenced libraries. The deactivation implementation shall be limited to destroy the object in terms of releasing memory resources. The activation and the deactivation shall not:

- throw errors,
- invoke any call-back to the FDI® Client; and
- invoke any user interaction.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

4.6 Interaction between an FDI® Client and a UIP

4.6.1 Handling of standard UI elements

UIPs shall delegate the presentation and handling of standard UI elements to the FDI® Client. The standard UI elements are

- UI Actions with standardized semantics (Apply/Close/Online Help), and
- specific UI Actions (UI actions, which are specific to one UIP, e.g. Reset, Reconnect).

To ensure a consistent user interface interaction across UIPs from different vendors, a UIP may delegate presentation and handling of additional UIP specific actions to the FDI® Client. Nonetheless UIPs are allowed to implement non-standard UI actions within their own UI area.

The set of standard UI actions and their respective semantics is fixed. However, the availability of these actions may change at any time depending on the internal state of the UIP. The set of additional UIP specific actions and their individual availability is not fixed. A UIP may add, remove, rename, enable or disable the UIP specific actions at any time depending on its requirements. The UIP has to inform the FDI® Client whenever the availability of its standard actions or UIP specific actions changes (see methods `Fdi.HostingServices.StandardActionItemSetChanged()` and `Fdi.HostingServices..ApplicationSpecificActionItemSetChanged()`).

An FDI® Client may use dedicated UI elements, e.g. button controls, to provide direct access to the standard actions, as well as indirectly invoke them in the context of user interaction with other FDI® Client UI elements. FDI® Client shall always show all custom actions exposed by a UIP with dedicated UI elements.

4.6.2 Non-blocking service execution

4.6.2.1 FDI® Client internal functions

The productive (time consuming) part of a service named *OperationName* shall be performed asynchronously by the FDI® Client. For the asynchronous service execution, the async/await pattern based on Promises is used. Therefore, any method of the FDI® interface implementation returns a Promise (see Figure 7).

If the request cannot be executed, the FDI® Client shall release the Promise with "reject" to indicate that the requested operation didn't start. Any other response shall release the Promise with "resolve". The return value provided with the "resolve" shall include an appropriate error message and one of the status codes defined with `Fdi.Model.StatusCode`, if the time consuming operation resulted in an error or has been cancelled.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023

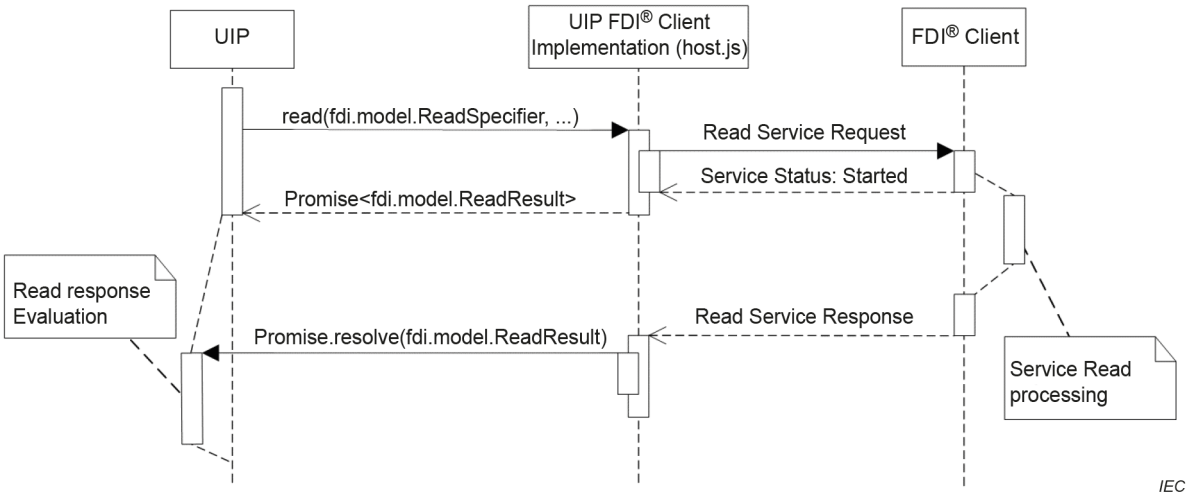


Figure 7 – Asynchronous service execution example (FDI® Client)

The implementation of Cancel*OperationName* is not implemented as separate service call. Instead, service calls that can be cancelled have an argument `cancelToken` of type `Fdi.Model.CancelToken`. If the UIP cancels the service execution, it calls the method `cancelToken.cancel()` which shall invoke the resolve of the Promise with `Fdi.Model.StatusCode.Bad_RequestCancelled` and an appropriate error message (see Figure 8).

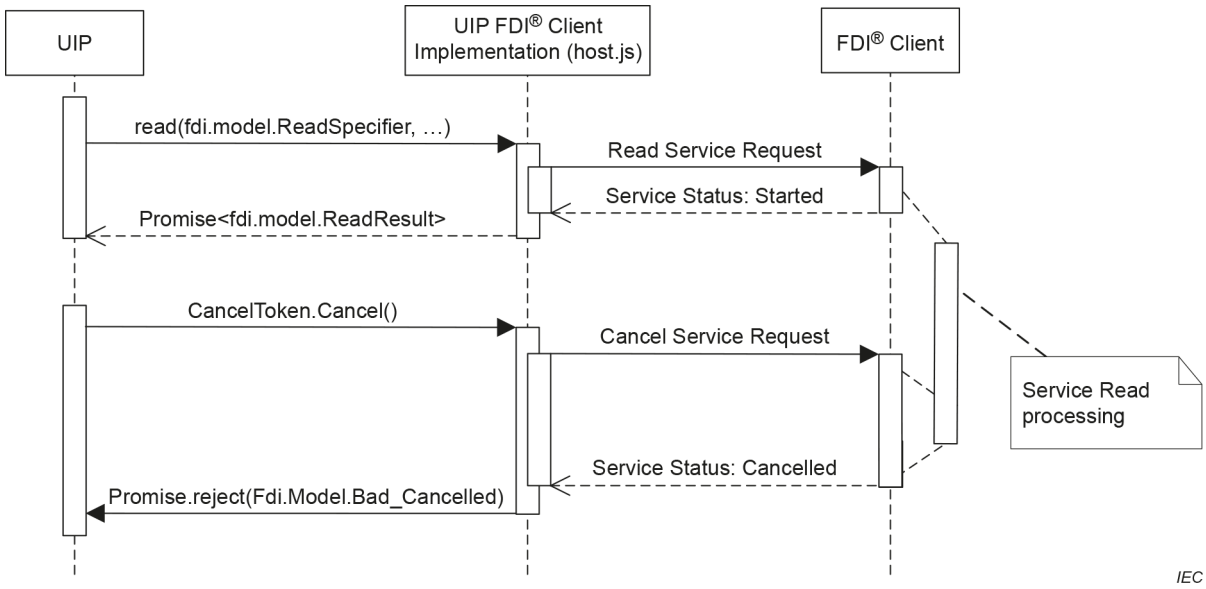


Figure 8 – Cancelled asynchronous service execution example (FDI® Client)

The handling of the operation within the UIP is described in the following 4.6.2.2.

4.6.2.2 UIP internal functions

The implementation of every service interface function that is called from the UIP, returns an asynchronous object of class Promise<OperationResult>". The UIP evaluates the Promise by implementing its ".then"-handler. In addition the ".catch()"-Handler should be implemented to catch unexpected errors. See example illustrated in Figure 9.

Documents
Document Sharing Hub
Full document details: IEC 62769-6-200-2023