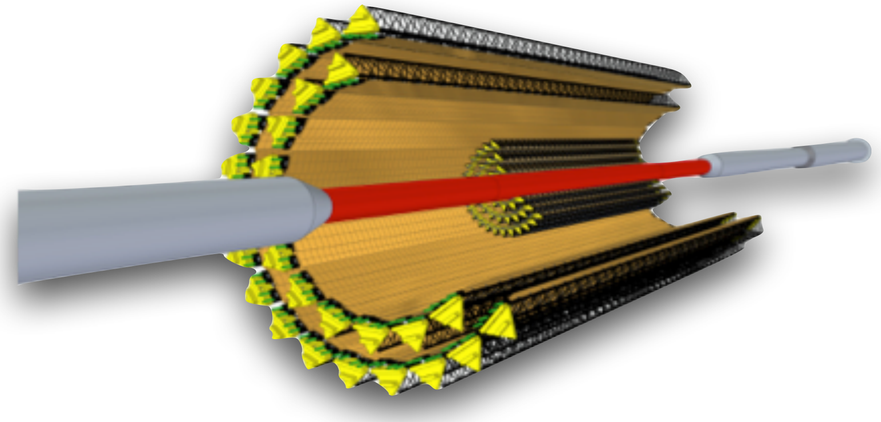
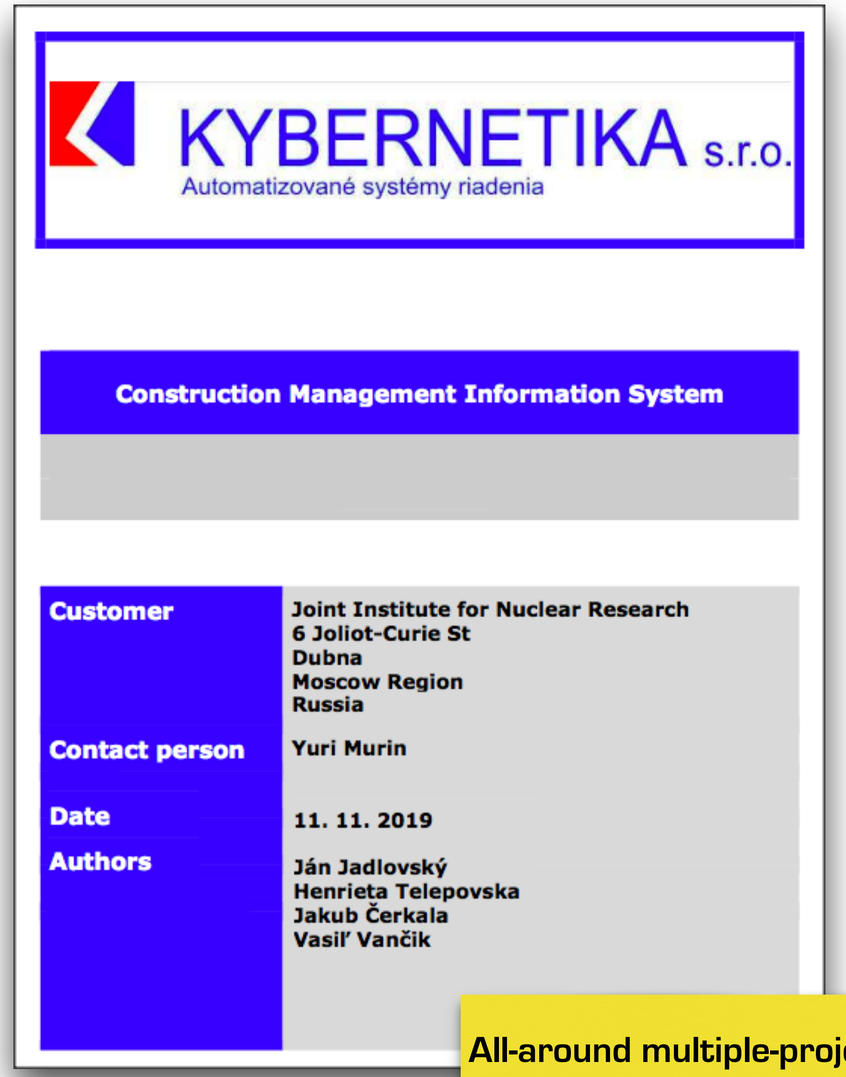
CMIS for the NICA projects of the STS Department.

César Ceballos Sánchez - JINR



JINR - VBLHEP, 06/02/2020

CMIS by Kybernetika



All-around multiple-projects handler.

It is composed of the following base modules:

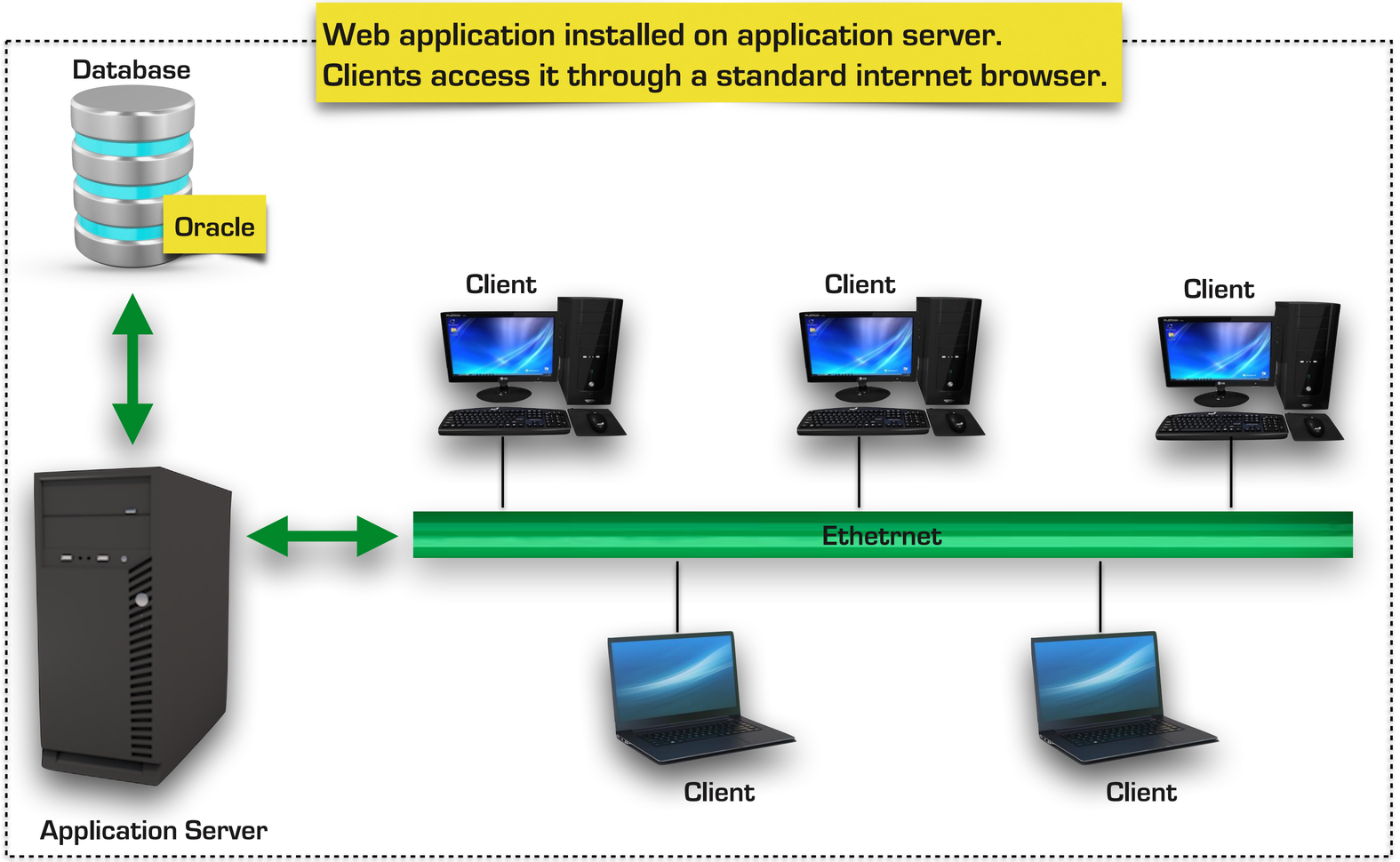
- **Resources Module**
  - Members
  - Institutes
- **Project Module**
  - Project Definition and Planning
  - Project Management and Resources
  - Construction Data
  - Activities
  - Finances
  - Reports
- **Administration**
- **Help**

CMIS by Kybernetika

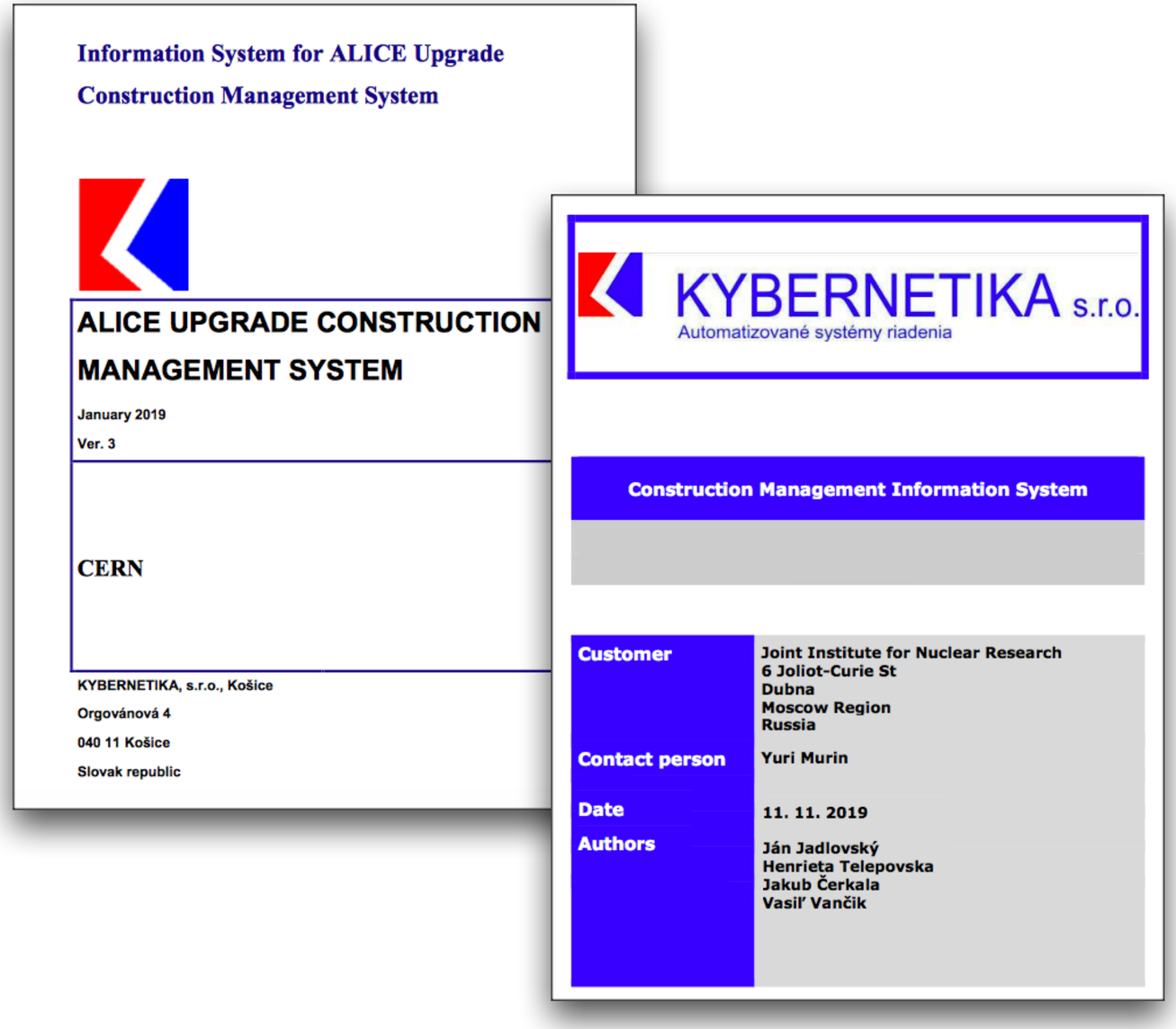
KYBERNETIKA s.r.o.
Automatizované systémy riadenia

Construction Management Information System

Customer	Joint Institute for Nuclear Research 6 Joliot-Curie St Dubna Moscow Region Russia
Contact person	Yuri Murin
Date	11. 11. 2019
Authors	Ján Jadlovský Henrieta Telepovska Jakub Čerkala Vasiľ Vančik



CMIS from ALUCMS by Kybernetika



Information System for ALICE Upgrade

Construction Management System

ALICE UPGRADE CONSTRUCTION MANAGEMENT SYSTEM

January 2019

Ver. 3

CERN

KYBERNETIKA, s.r.o., Košice

Orgovánová 4

040 11 Košice

Slovak republic

KYBERNETIKA s.r.o.

Automatizované systémy riadenia

Construction Management Information System

Customer	Joint Institute for Nuclear Research 6 Joliot-Curie St Dubna Moscow Region Russia
Contact person	Yuri Murin
Date	11. 11. 2019
Authors	Ján Jadlovský Henrieta Telepovska Jakub Čerkala Vasiľ Vančik

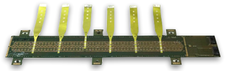
ALUCMS in ALICE-ITS Upgrade

OB Stave Production

Berkely (USA), Daresbury (UK), Frascati (Italy), Nikhef (Netherlands), Turin (Italy)

OB HIC Production

Bari (Italy), Liverpool (UK), Pusan/Inha (Rep. of Korea), Strasbourg (France), Wuhan (China)

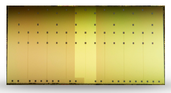


OB FPC Production

SWISS GS, Hybrid SA (Switzerland), Catania, Trieste (Italy)

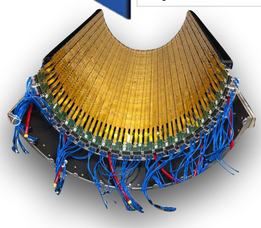
Chip Test

Pusan/Inha, Yonsei (Rep. of Korea)

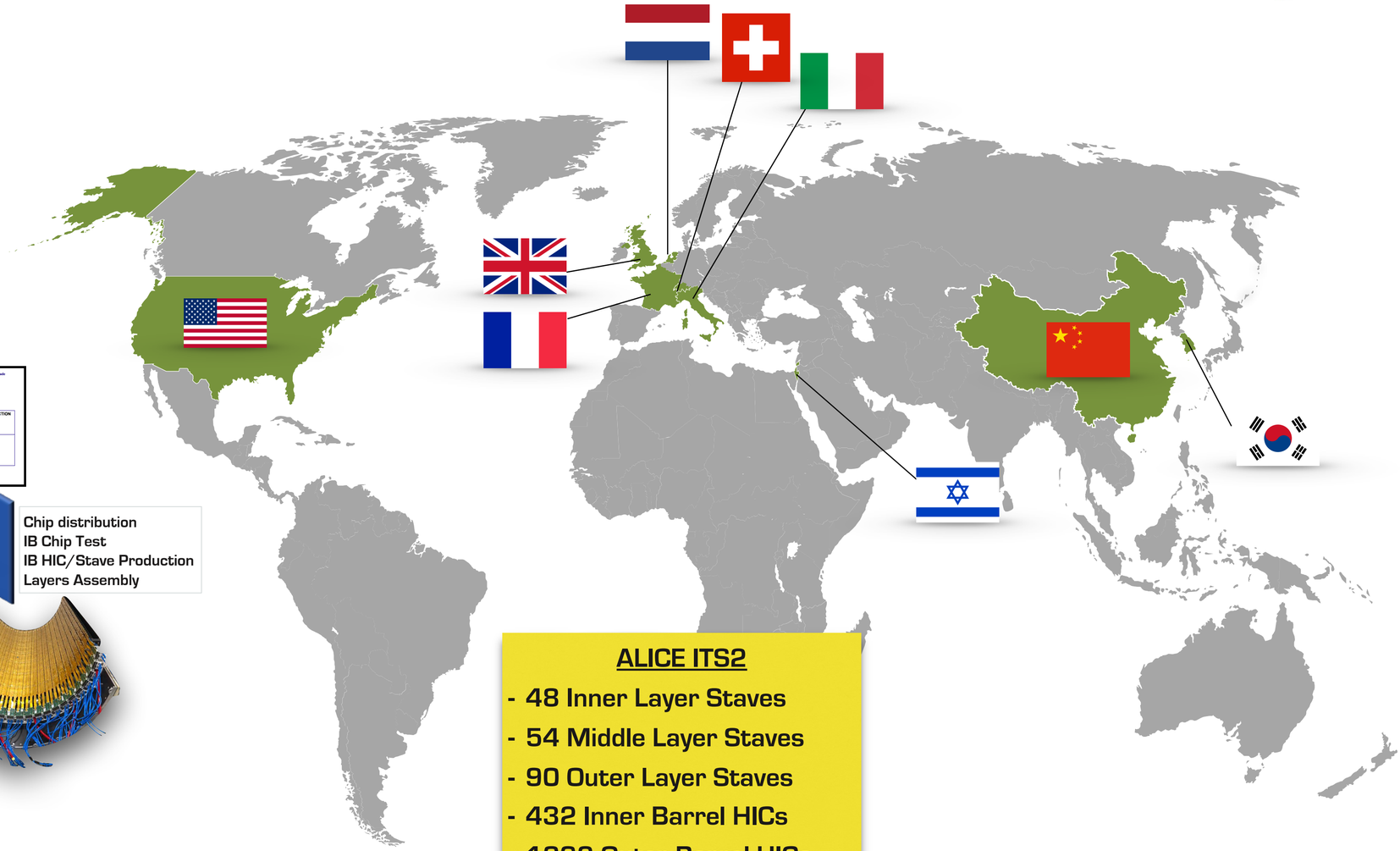


Chip production

MEMC (Italy), TowerJazz (Israel), Furex (Rep. of Korea)



Chip distribution
IB Chip Test
IB HIC/Stave Production
Layers Assembly



ALICE ITS2

- 48 Inner Layer Staves
- 54 Middle Layer Staves
- 90 Outer Layer Staves
- 432 Inner Barrel HICs
- 1692 Outer Barrel HICs
- 27576 Pixel Chips

(plus spares)

CMIS full structure

- **Resources**
  - Institutes
  - Members
- **Projects**
  - Project A
  - Project B
  - Project C
- **Administration**
  - Roles
  - Users
  - Att. Categories
  - Positions
  - Qualifications
- **Settings**
  - Management Functions
  - Activity Types
  - Activity Status
  - Parameters
  - Component Categories
  - Component Types
  - Component Type Graph
  - Raw Material Types
  - Units
- **Resources**
  - Institutes
  - Members
  - Suppliers
  - Suppliers Contact
- **Organization**
  - Structure
  - Management
- **Planning**
  - Tasks (Gantt)
  - Milestones
  - Milestone Versions
- **Constr. Data**
  - Activities
  - Components
  - Raw Materials
- **Accountings**
  - Invoices
  - Contributions
  - WBS
- **Reports**
  - Components by AT

Outside the Projects

Project-specific

‘Settings’ & ‘Construction Data’

- Resources
  - Institutes
  - Members
- Projects
  - Project A
  - Project B
  - Project C
- Administration
  - Roles
  - Users
  - Att. Categories
  - Positions
  - Qualifications

Settings
- Management Functions
- **Activity Types**
- Activity Status
- Parameters
- Component Categories
- **Component Types**
- Component Type Graph
- Raw Material Types
- Units

Resources
- Institutes
- Members
- Suppliers
- Suppliers Contact

Organization
- Structure
- Management

Planning
- Tasks (Gantt)
- Management
- Milestones
- Milestone Versions

Constr. Data
- **Activities**
- **Components**
- Raw Materials

Accountings
- Invoices
- Contributions
- WBS

Reports
- Components by AT

Outside the Projects

Project-specific

Activities and Components

Activities are actions performed on Components that may change their Physical/Functional[*] status.

Activity: ALPIDEB Chip Classification of ChipXX

Component: ALPIDEB ChipXX

Result: ALPIDEB ChipXX [Physical Satus] = <physical_status_value>
ALPIDEB ChipXX [Functional Satus] = <functional_status_value>

[*] **All possible Physical and Functional statuses of a component are declared on the 'Component Type' definition.**
The possible result of an Activity are combinations of Physical and functional statuses and are declared on the 'Activity Type' definition.

Types and Instances of a Type

Type

Activity Type: ALPIDEB Chip Classification

Component Type: ALPIDEB Chip

Instance

Activity: Classification T854193W09R41

Component: T854193W09R41

General definitions

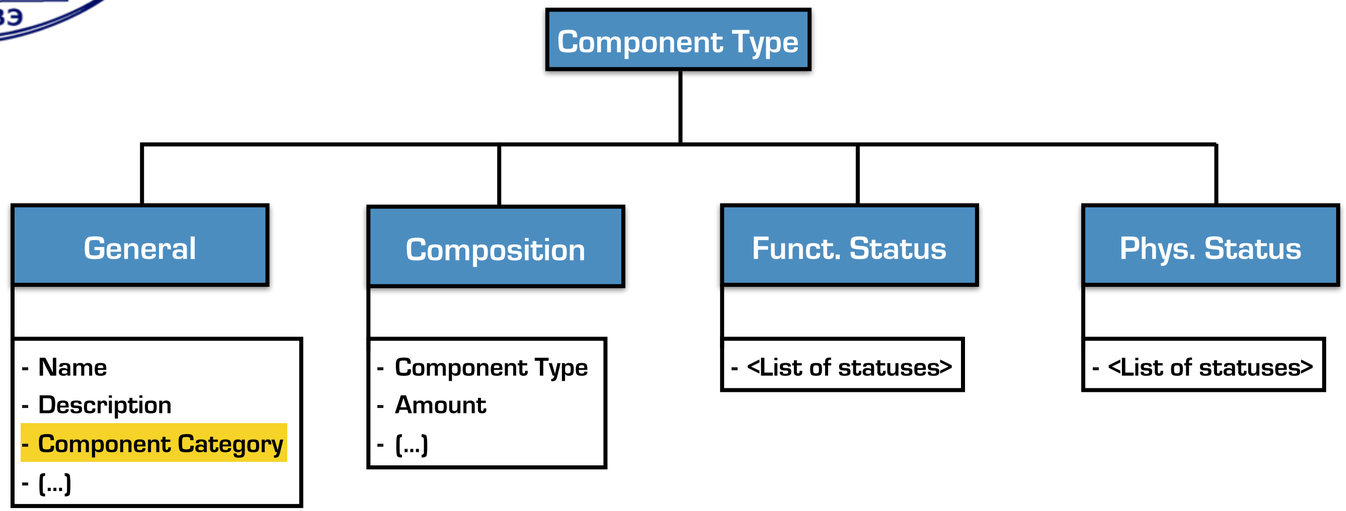
Activity Status: Open(default), Close(default),...

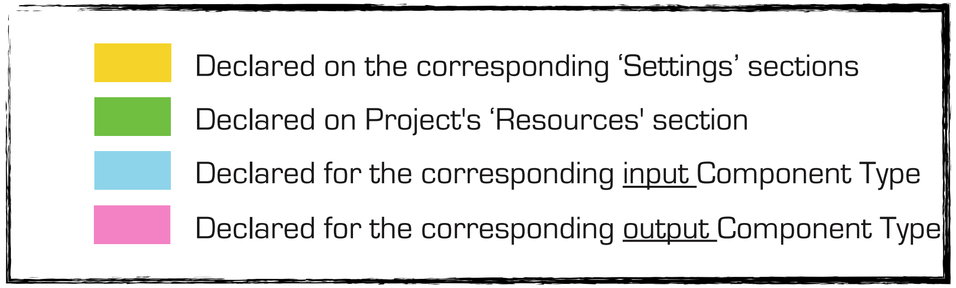
Component Category: Carbon support structure, Components container, ...

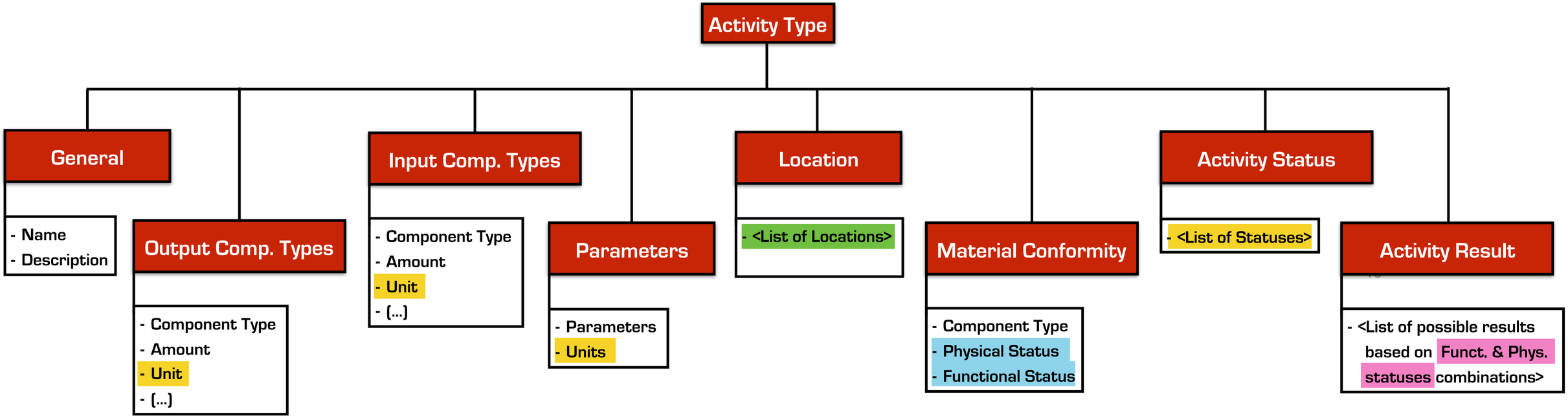
Parameters: Defect density, DVDD Impedance, ...

Units: spatial density [cm^(-2)], Volts [V], piece(s) [pcs], ...

Component Types and Activity Types main structure

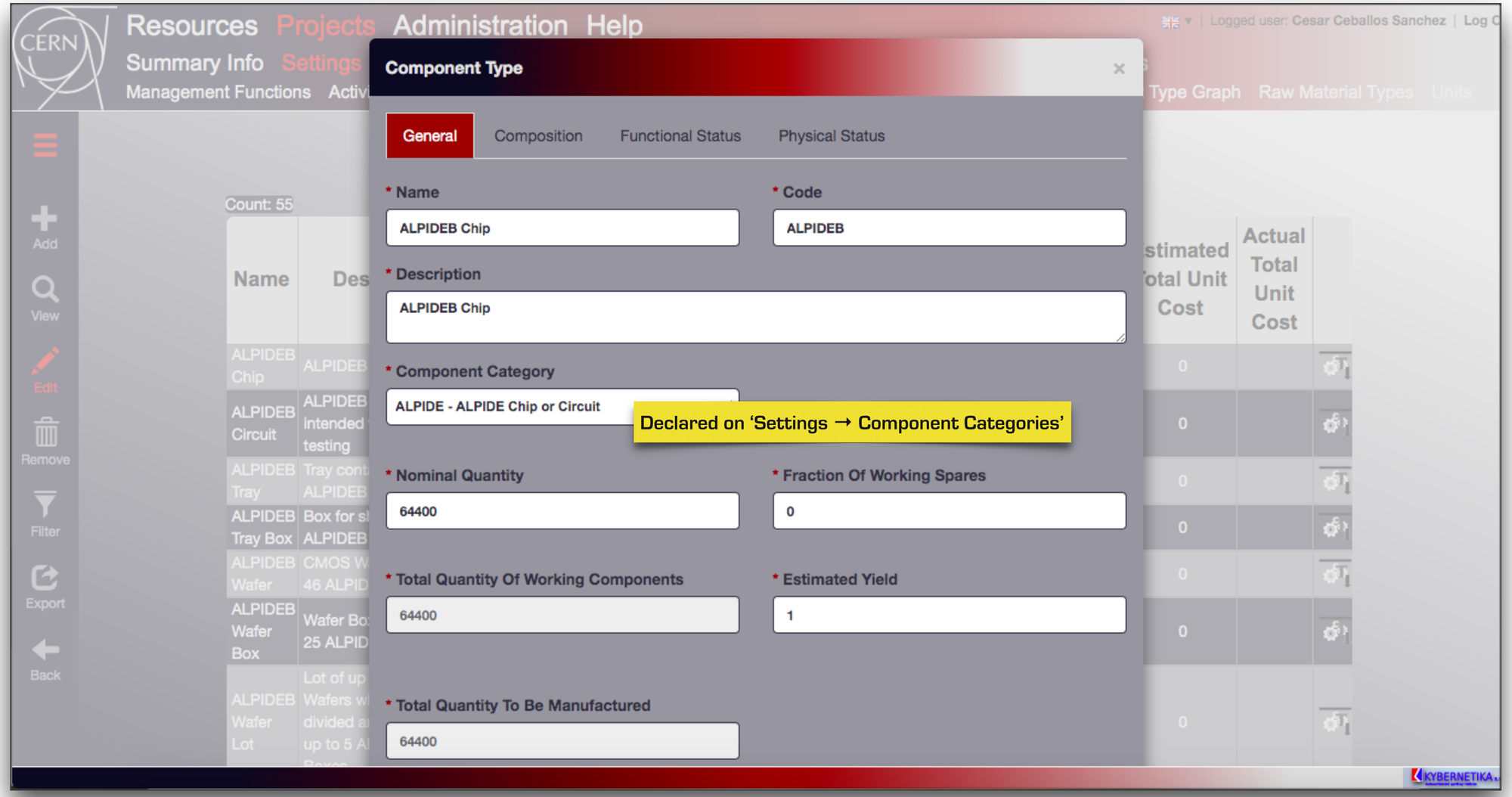






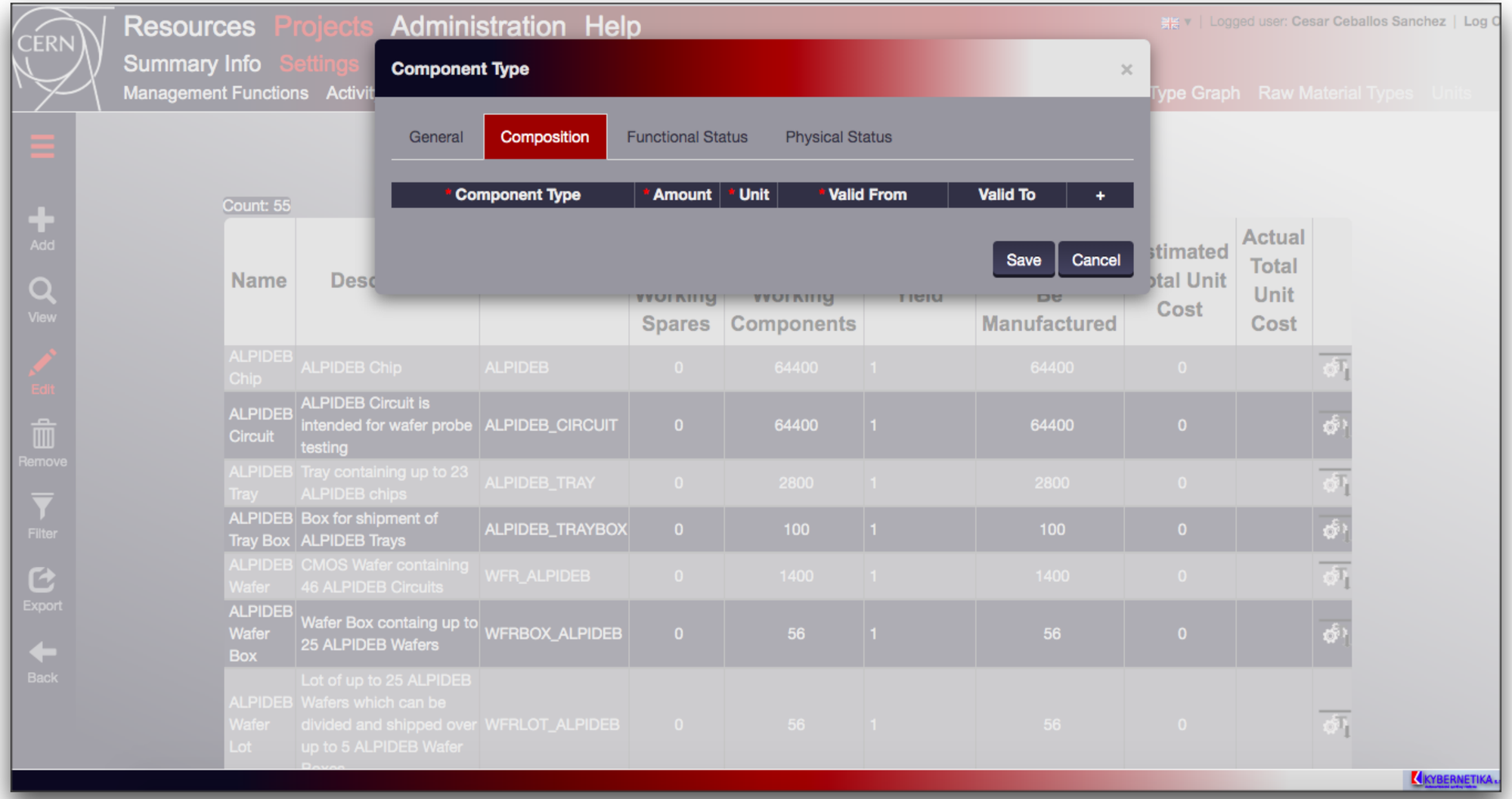
Component Type 'ALPIDEB Chip'

MPD - ITS



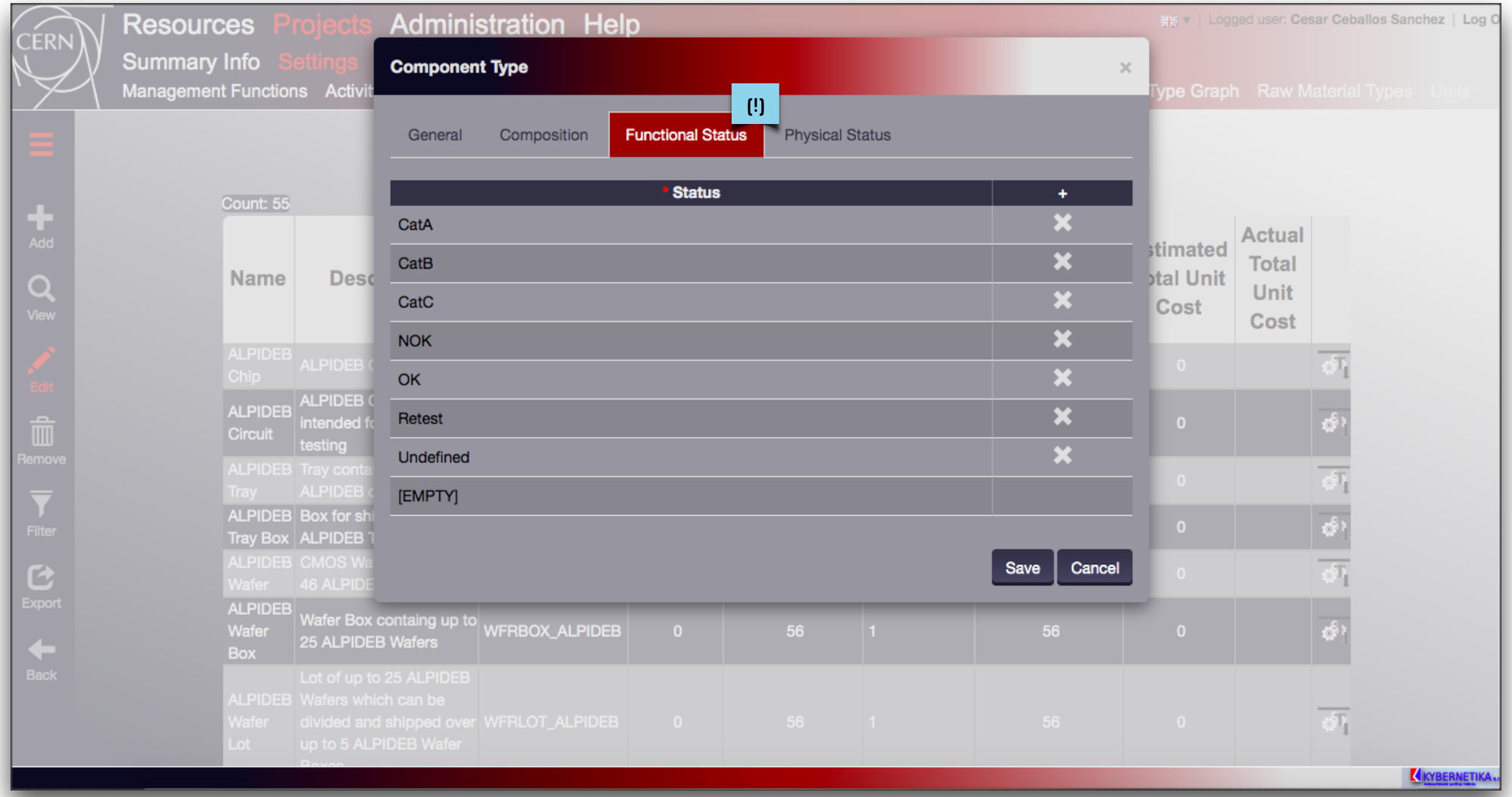
Component Type 'ALPIDEB Chip'

MPD - ITS



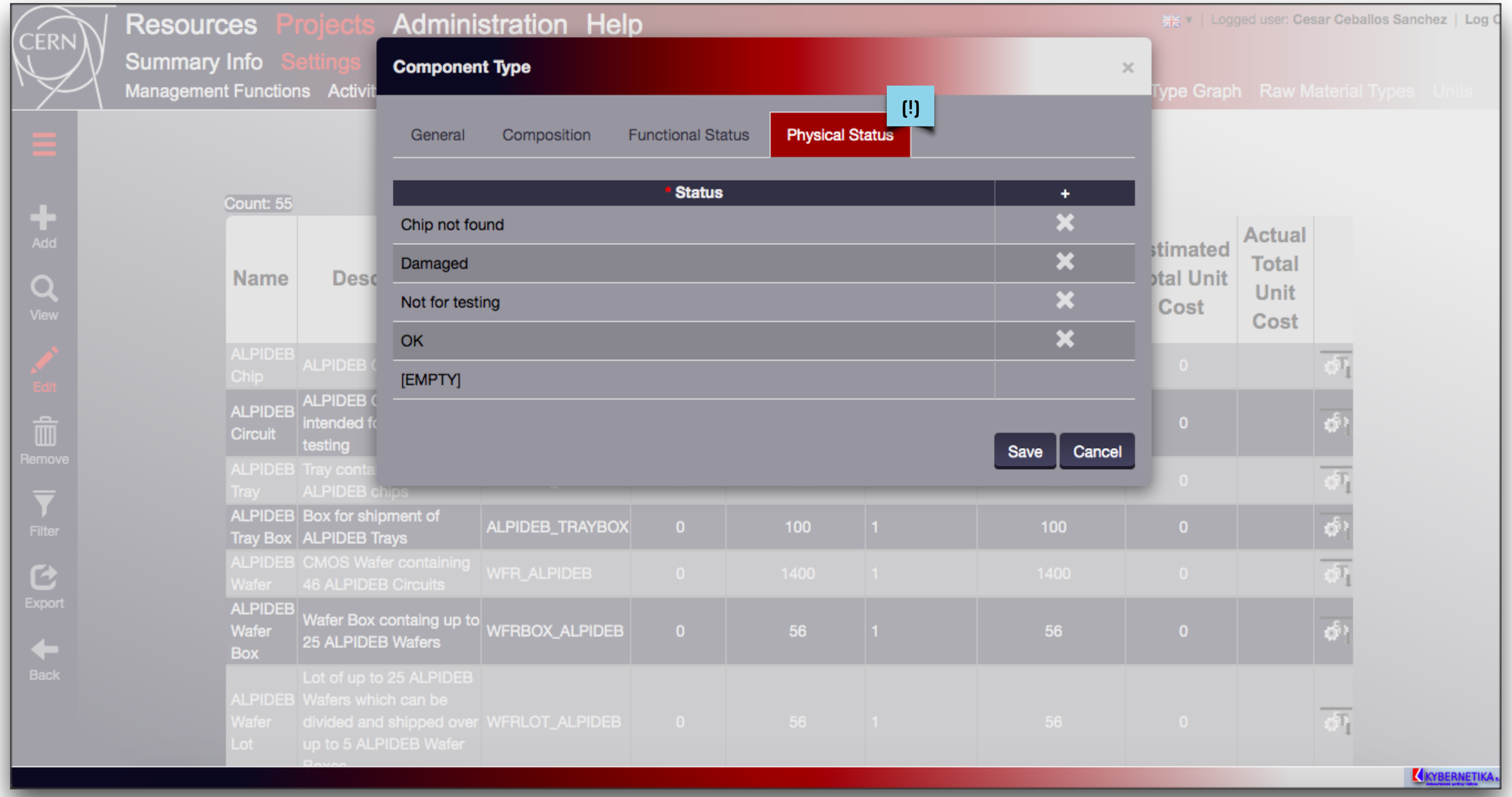
Component Type ‘ALPIDEB Chip’

MPD - ITS



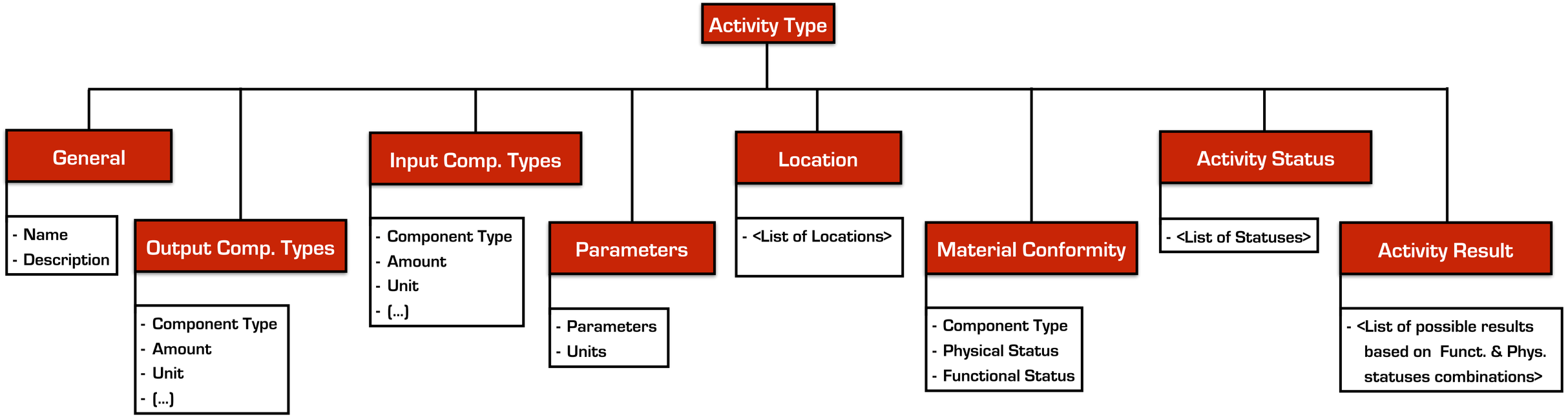
Component Type ‘ALPIDEB Chip’

MPD - ITS



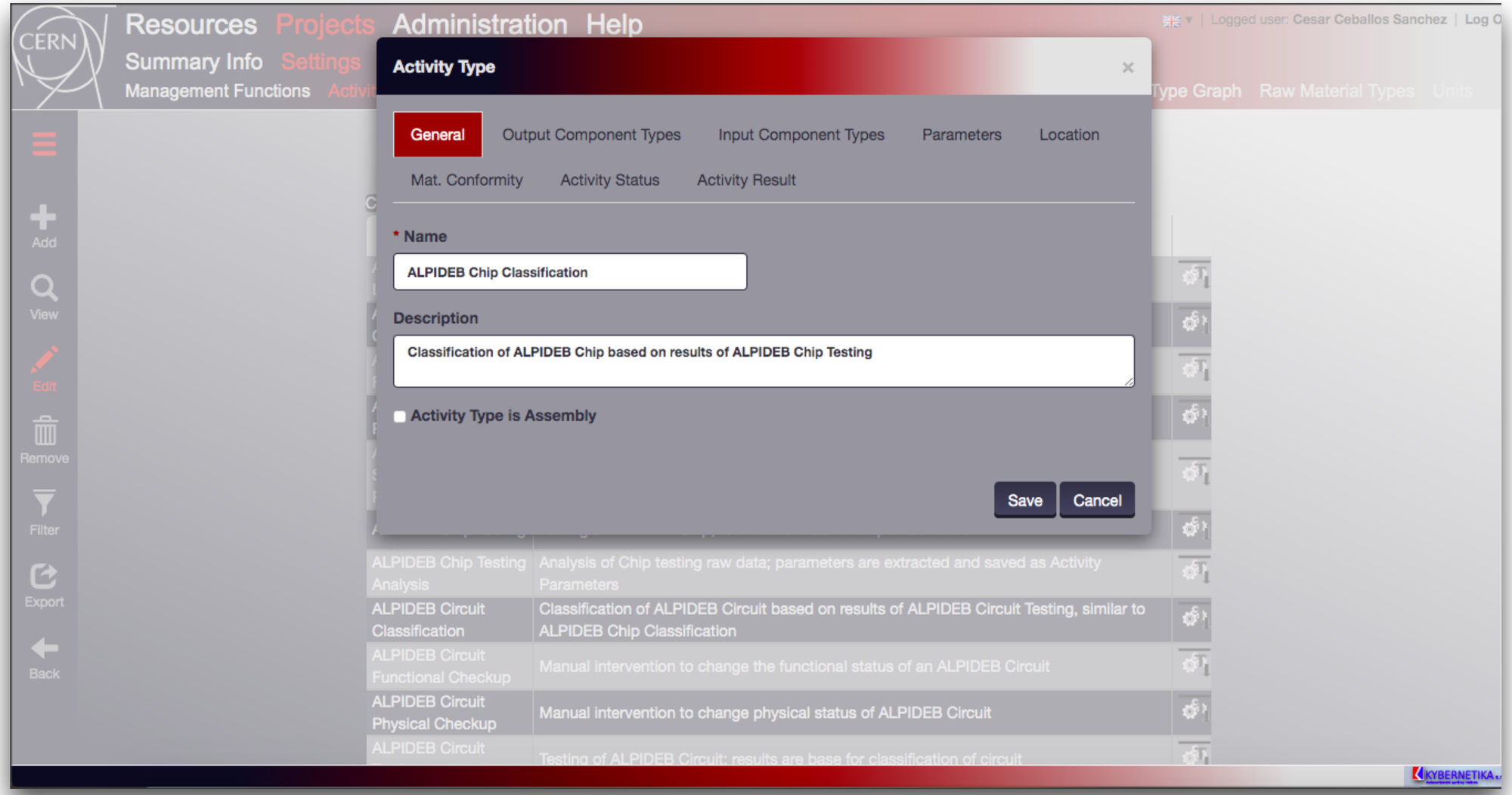
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



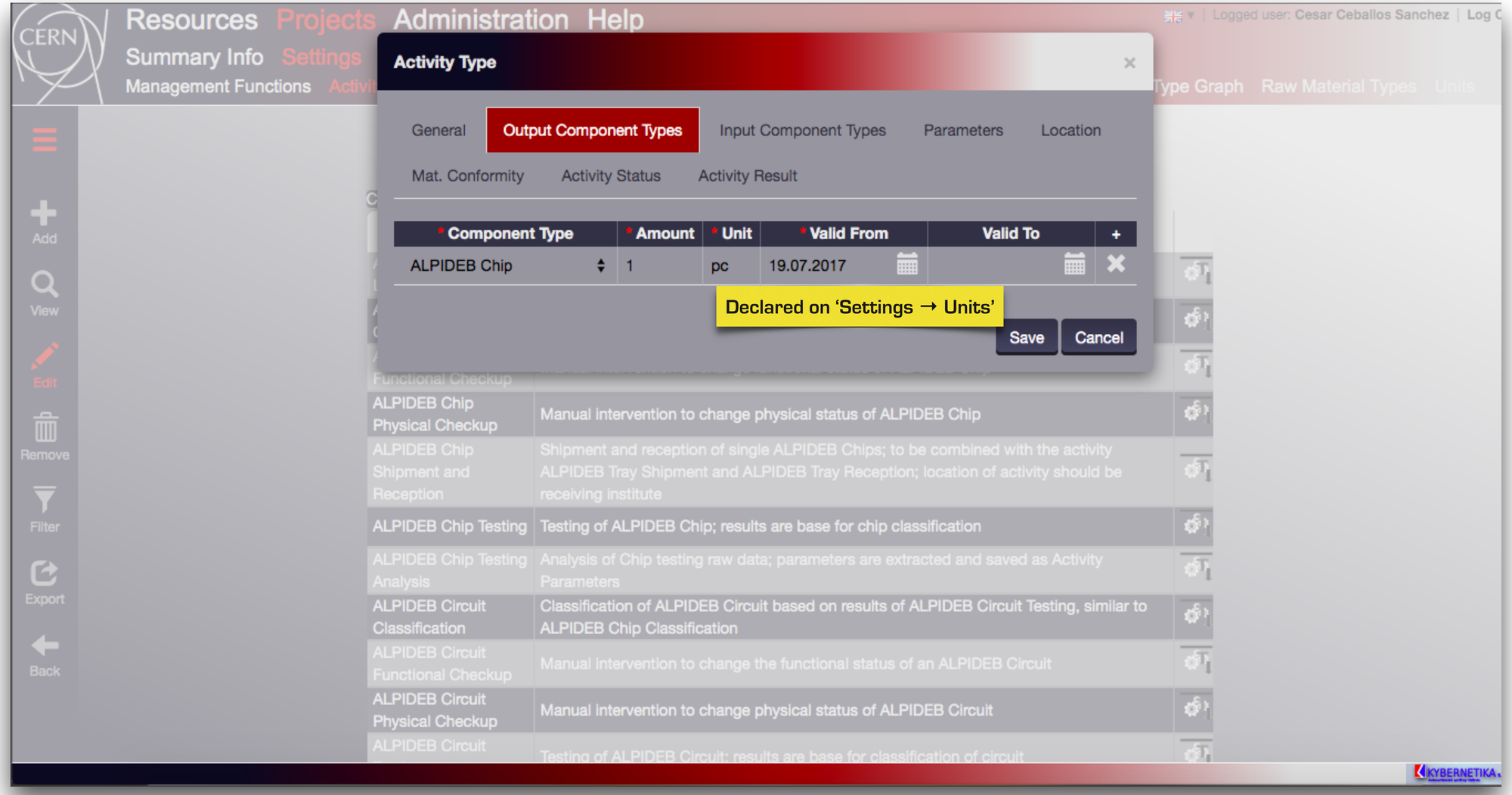
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



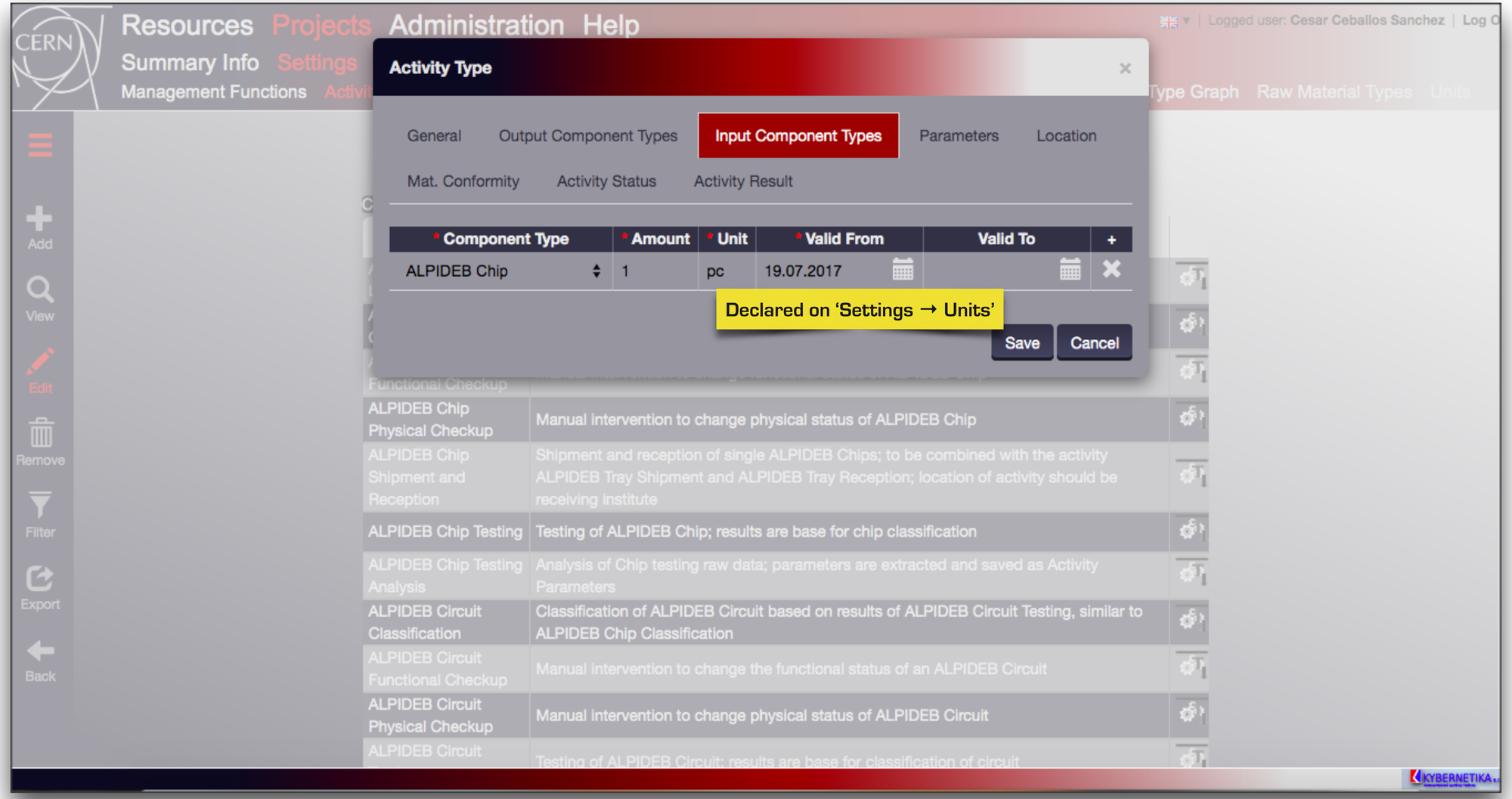
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



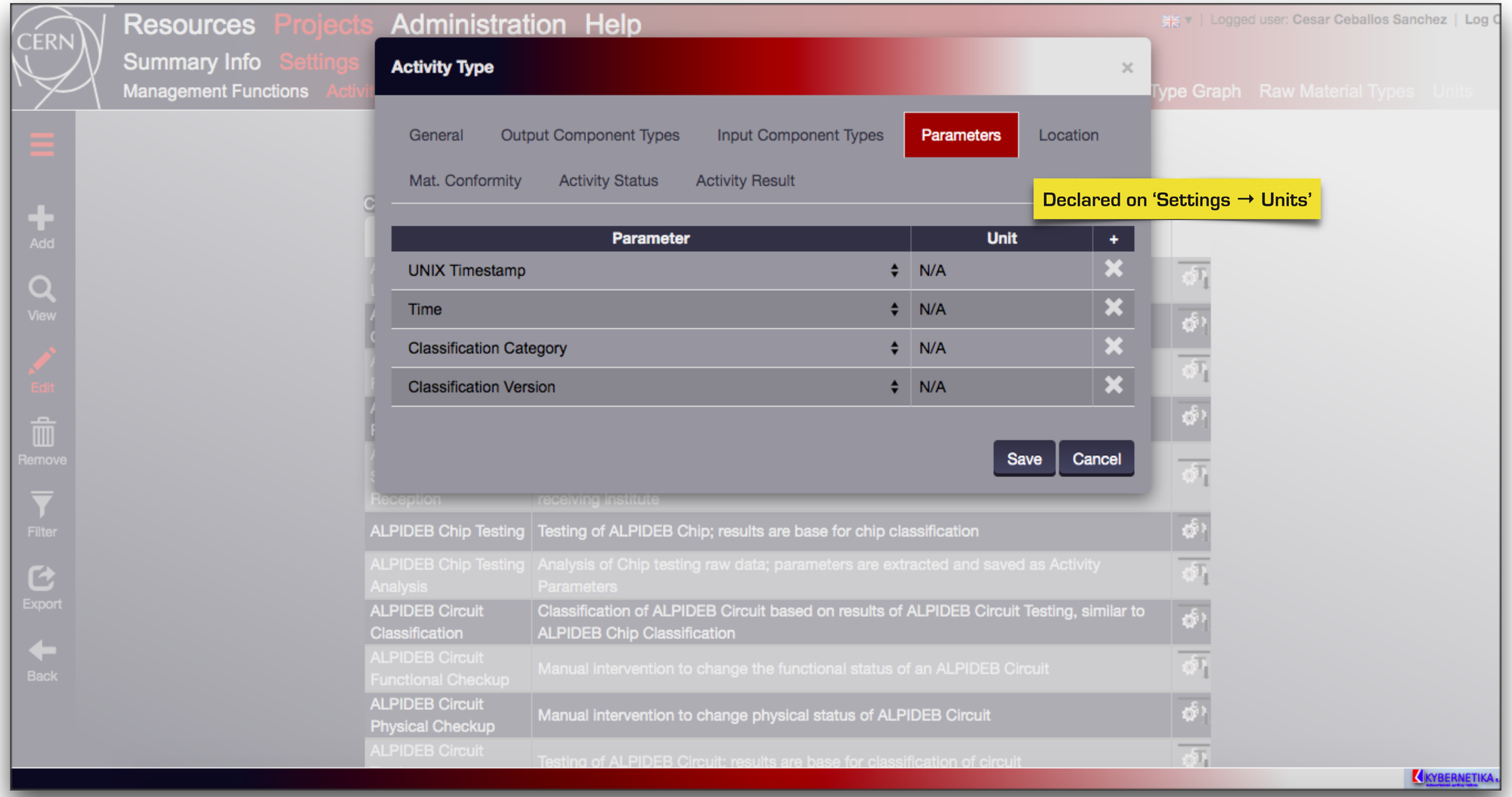
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



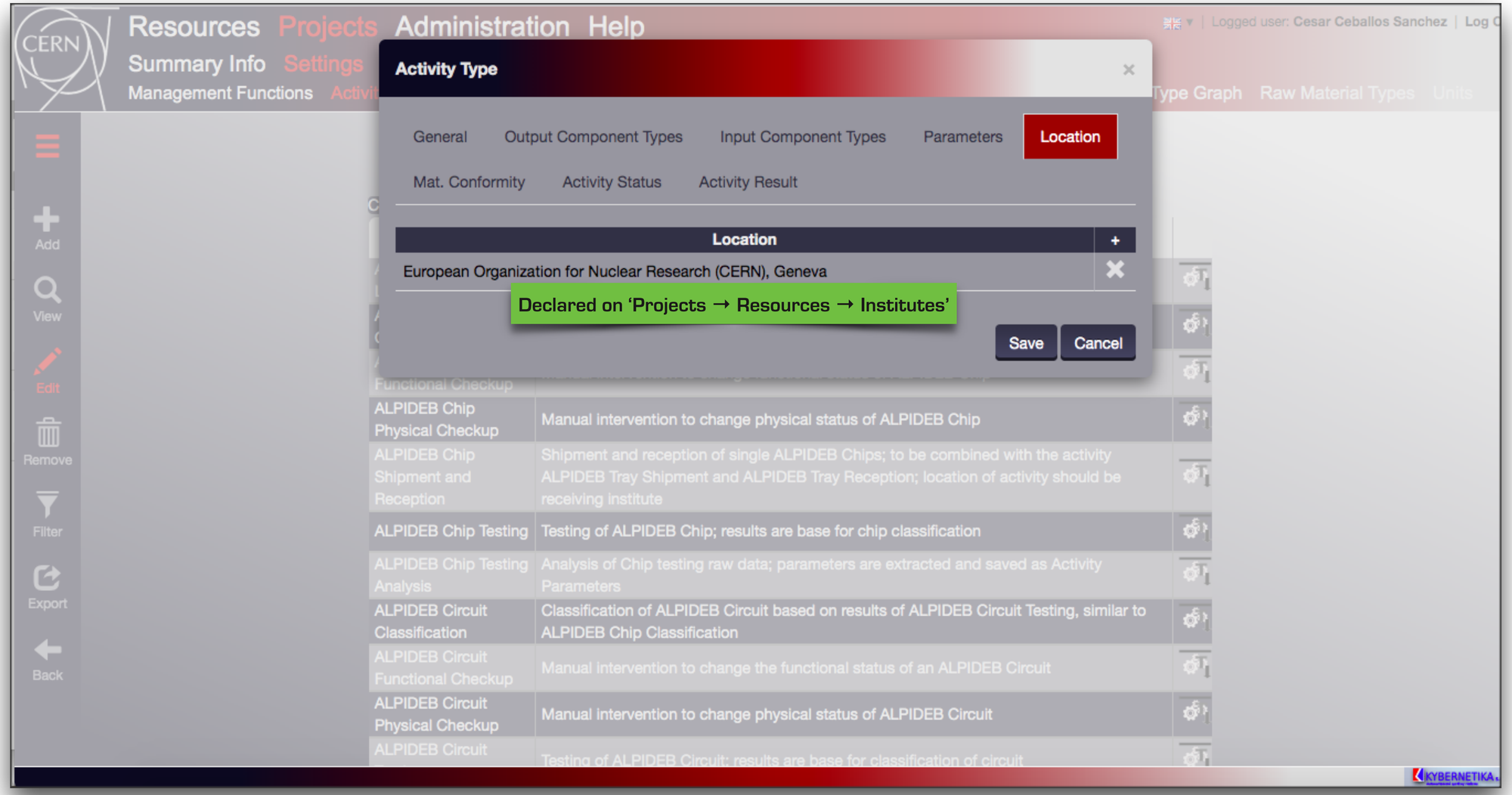
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



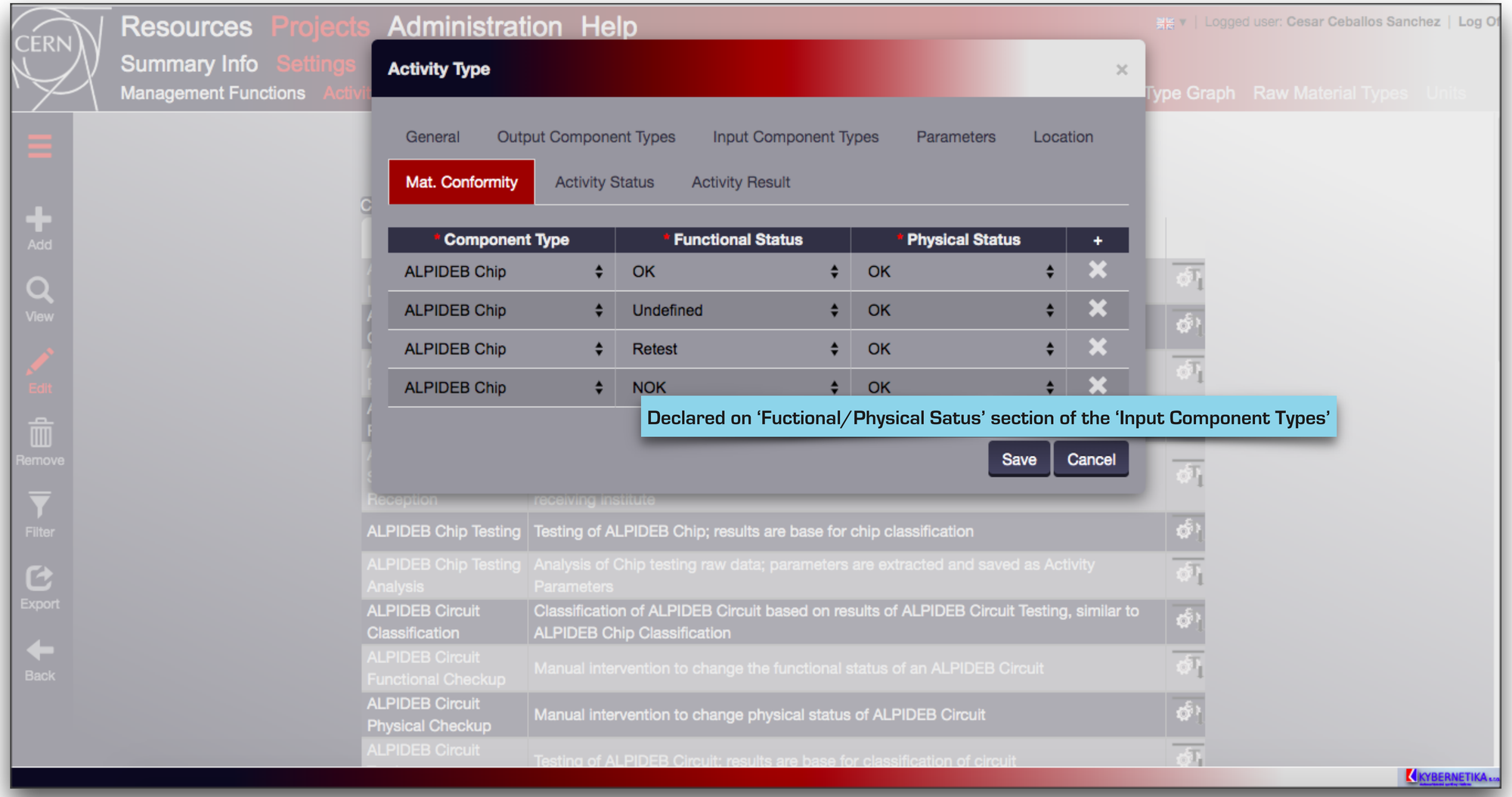
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



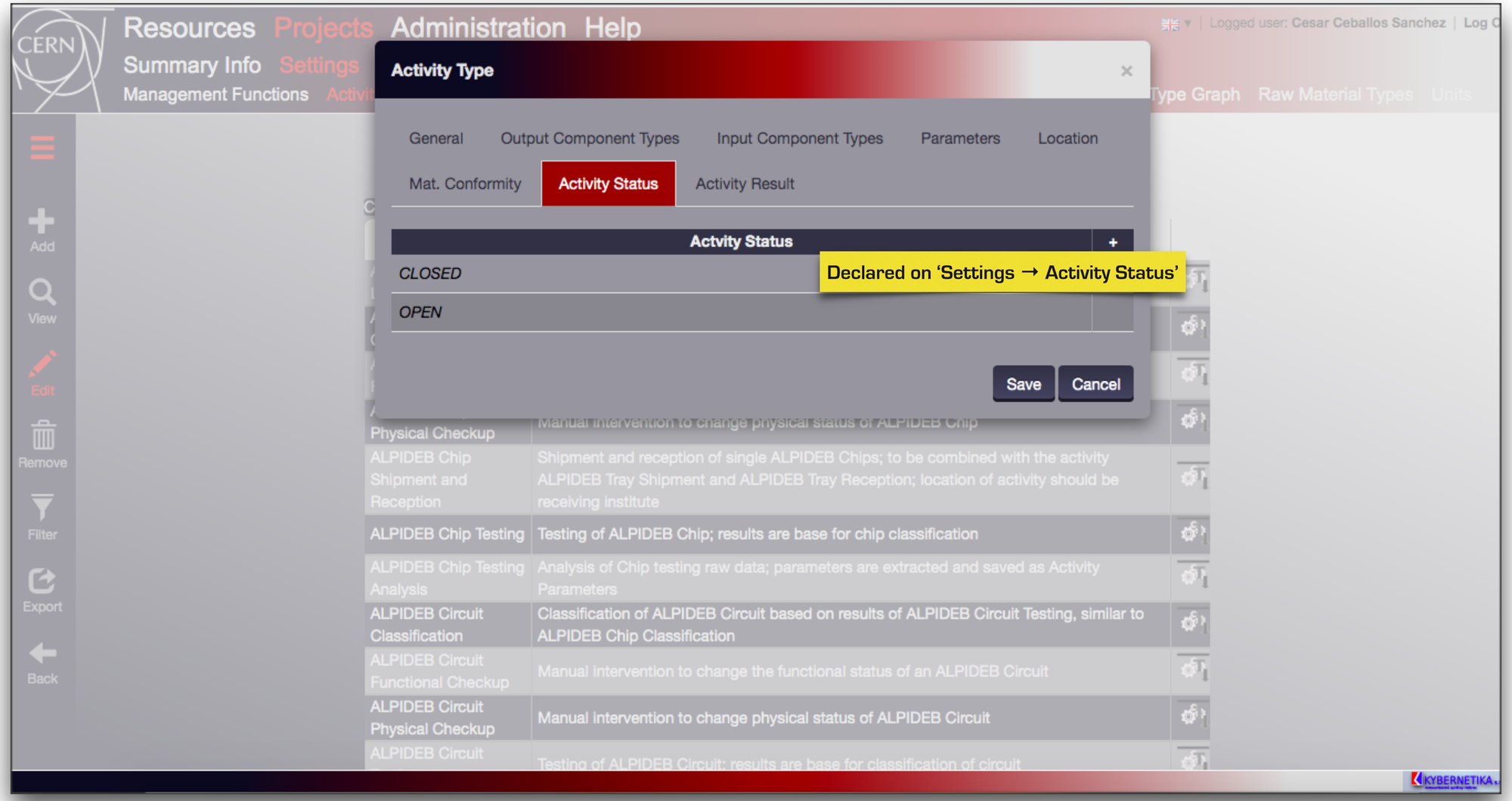
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



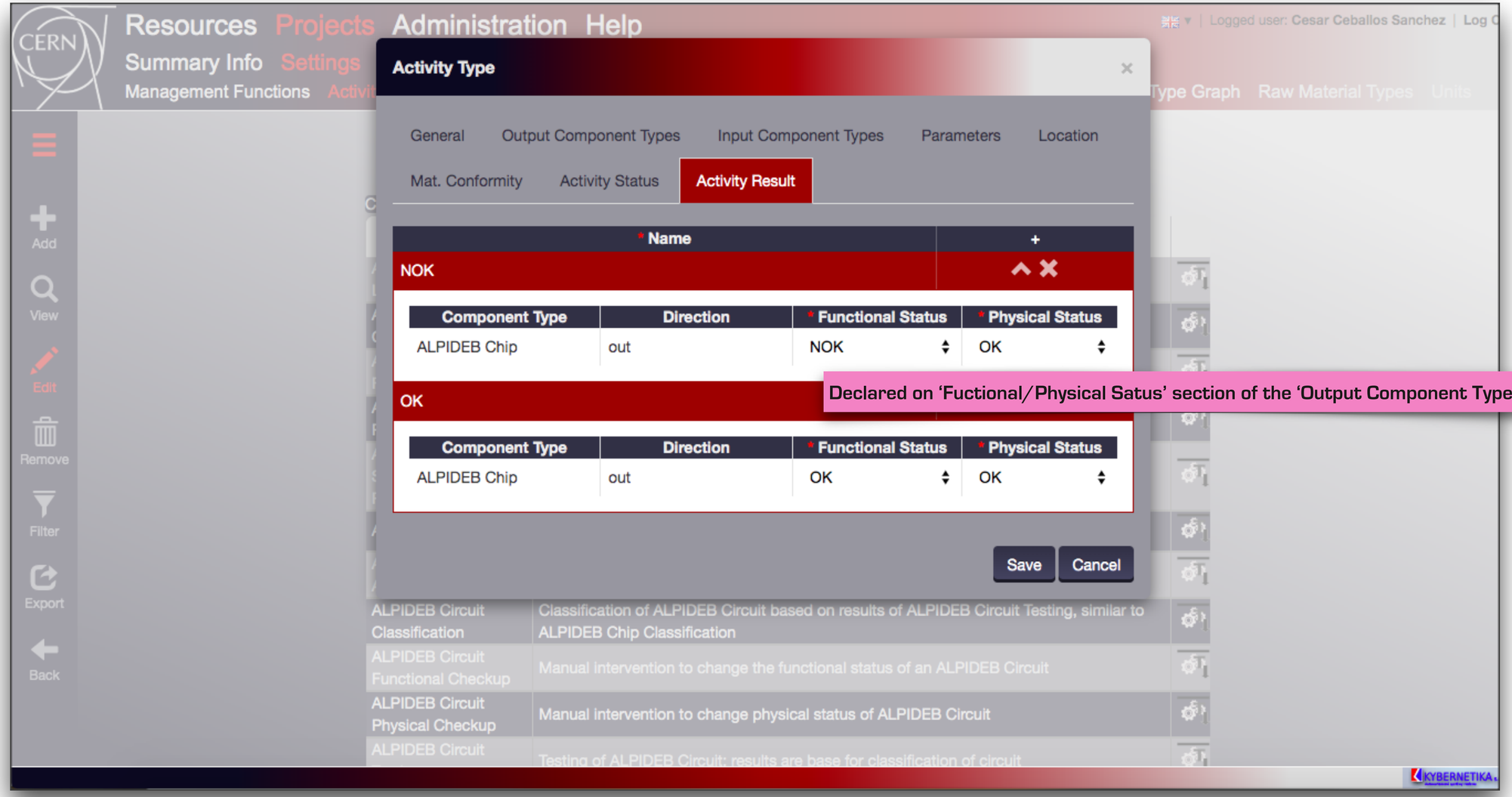
Activity Type ‘ALPIDEB Chip Classification’

MPD - ITS



Activity Type 'ALPIDEB Chip Classification'

MPD - ITS

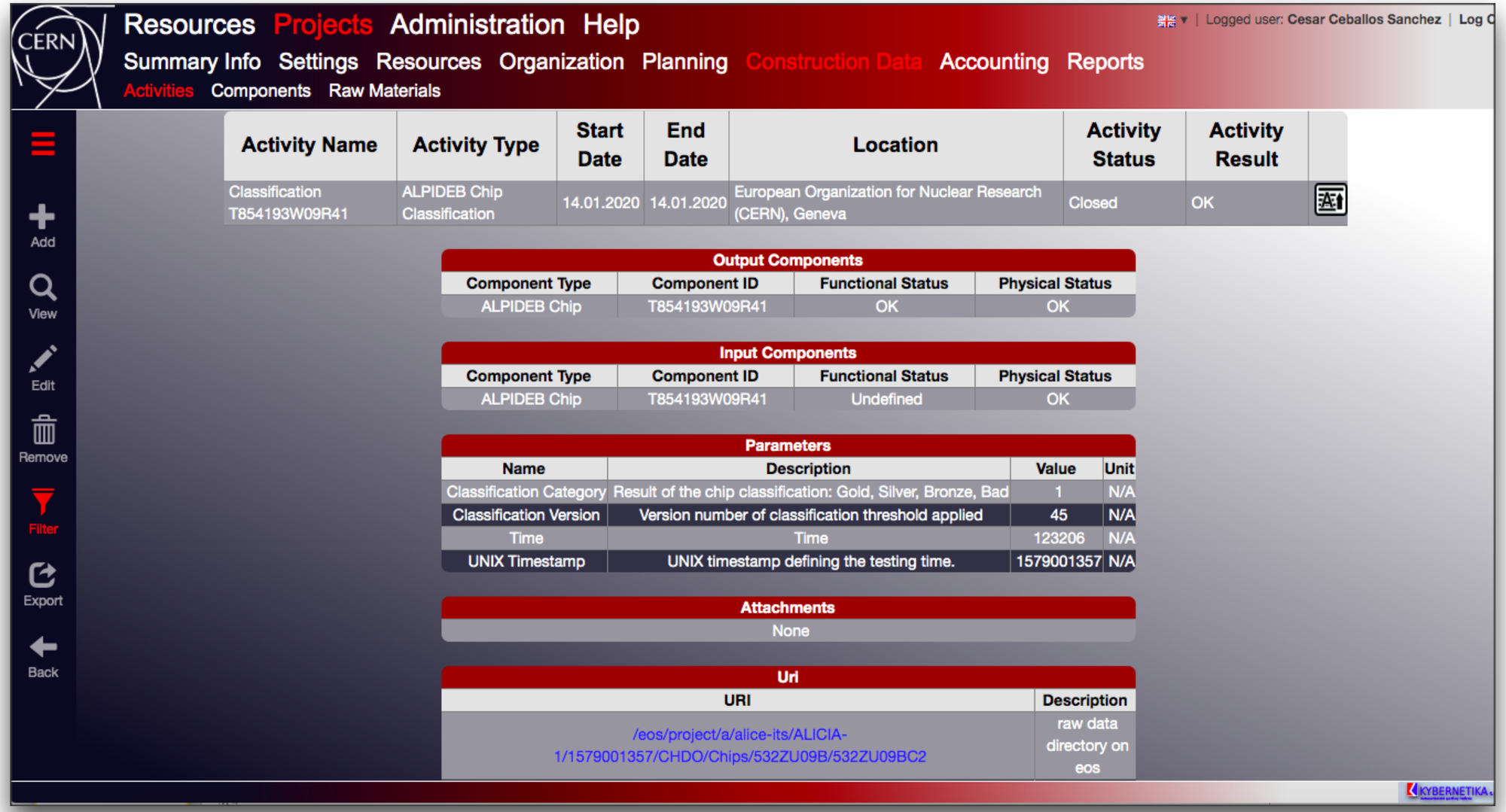


Declared on 'Fuctional/Physical Satus' section of the 'Output Component Types'

Create a specific activity

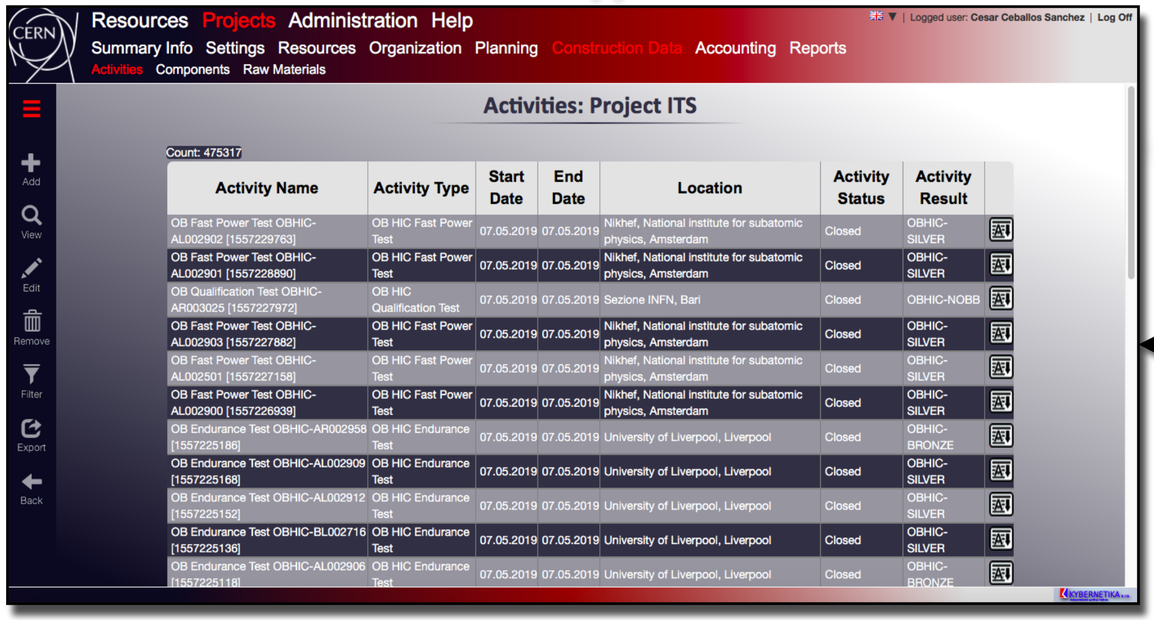
MPD - ITS

Activity ‘Classification T854193W09R41’



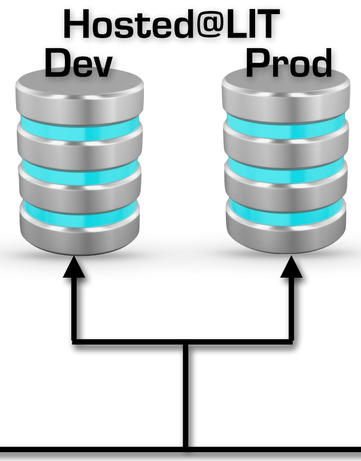
Web Interface & API functions

WebApp



Web interface to be used by registered users for reading/writing data from/to the DB.

Activity and Component Types definitions are always done via WebApp



API functions

AlucmsWebAPI

The following operations are supported. For a formal definition, please review the Service Description.

- ActivityAttachmentCreate
- ActivityAttachmentRemove
- ActivityChange
- ActivityComponentAssign
- ActivityComponentRemove
- ActivityCreate
- ActivityMemberAssign
- ActivityMemberChange
- ActivityMemberRemove
- ActivityParameterChange
- ActivityParameterCreate
- ActivityParameterRemove
- ActivityRead
- ActivityReadOne
- ActivityTypeRead
- ActivityTypeReadAll
- ActivityUriChange
- ActivityUriCreate
- ActivityUriRemove
- AttachmentCategoryRead
- ComponentActivityHistoryRead
- ComponentChange
- ComponentChildrenRead

API function collection meant to be used by the construction and test hardware to read/write data from/to the DB.

Foreseen DB system implementation at JINR (LIT)

CMIS

(Oracle)

File attachments

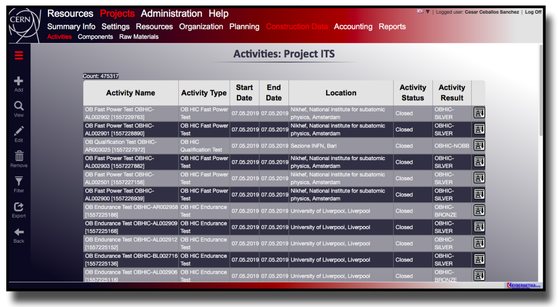
EOS(*)

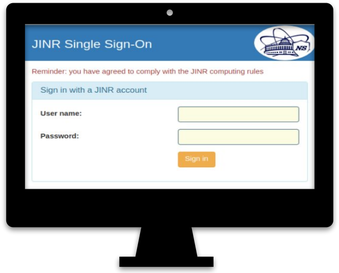
webApp
(HTTPS)

API functions
(HTTP GET/POST & SOAP)

(SSH)

(SSO cookie)



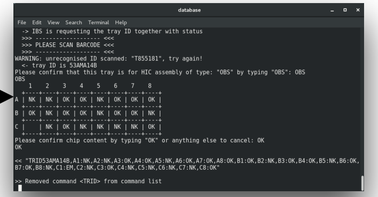


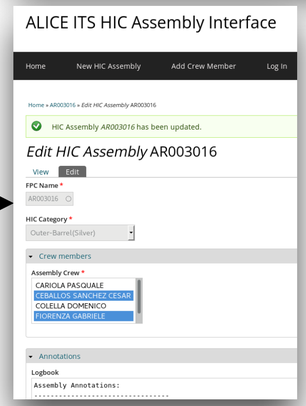
JINR SSO:

- Web access.
- Role-based access rights.

Custom Assembly & Testing programs

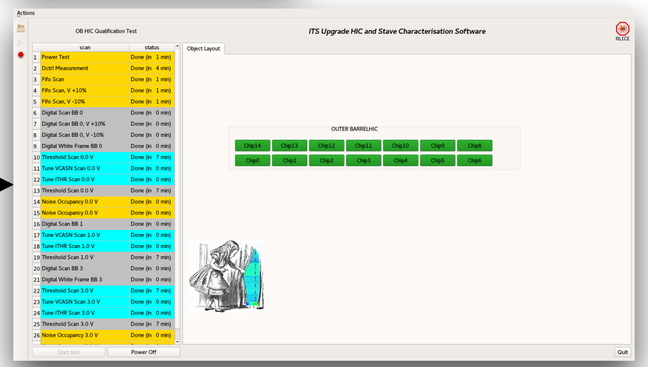
(Kerberos credentials automatically renewed)





JINR SSO:

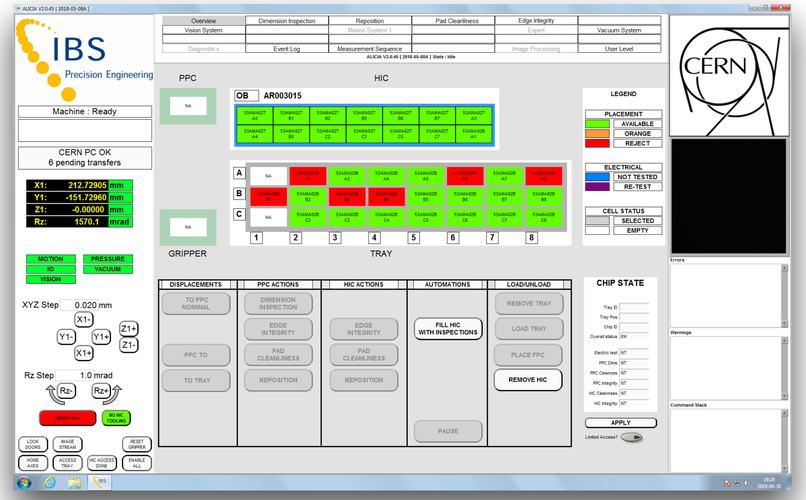
- Registered Service accounts
- Kerberos credentials
- Access cookies

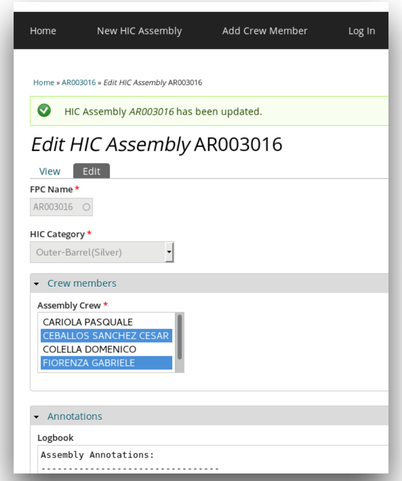


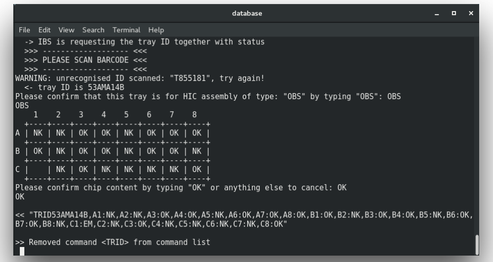
(*)Network File System

OBHIC Assembly - Using the API functions

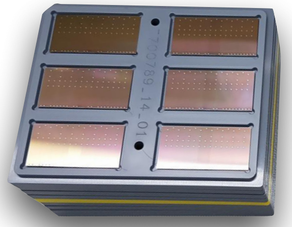
MPD - ITS





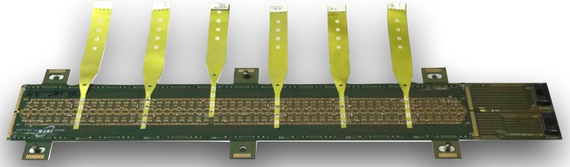


2 x 7 Chips



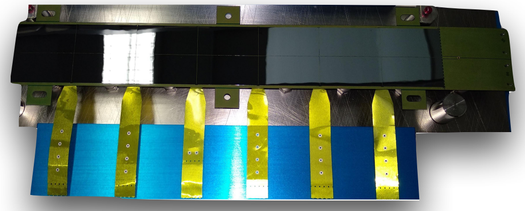
+

1 FPC



=

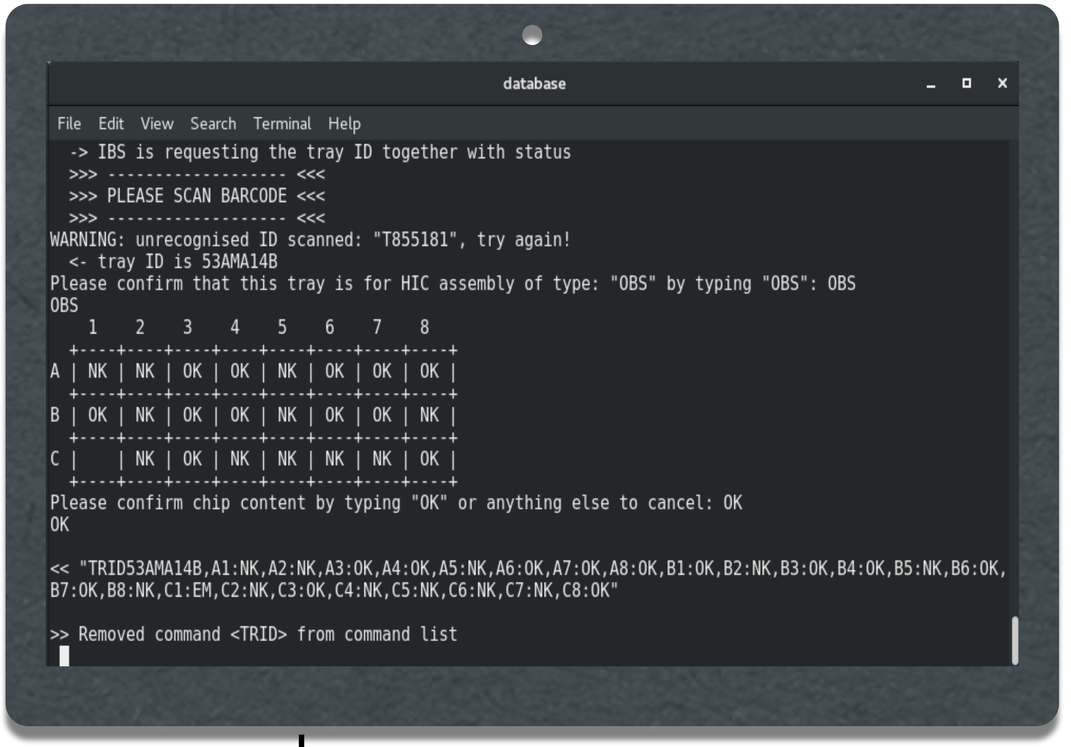
1 OBHIC



Outer Barrel HIC assembly - Chips selection

MPD - ITS

Parameters			
Name	Description	Value	Unit
Analog Current	Analog Current consumption of e.g. ALPIDEB	11.6233333333	mA
Analogue Scan 3V Broken Pixels		4	N/A
Analogue Scan 3V Dead Pixels		4274	N/A
Analogue Scan 3V Noisy Pixels		30	N/A
BackBias 3V PWELL/SUB Current		2.56843	mA
BackBias Avg Analog Current		16.5051851852	A
BackBias Avg Digital Current		57.1281481481	A
BackBias Avg PWELL/SUB Current		2.63092111111	A
BackBias Max Analog Current		115.66	mA
BackBias Max Digital Current		61.46	mA
BackBias Max PWELL/SUB Current		3.55672	mA
BackBias Max Voltage	Maximum value of back bias possible to be applied to ALPIDEB	-5.2	V
Broken Pixels	Number of broken pixels in matrix	4	N/A



1. Scan tray's QR code
2. Read test results
3. Get tray map
4. Load chips

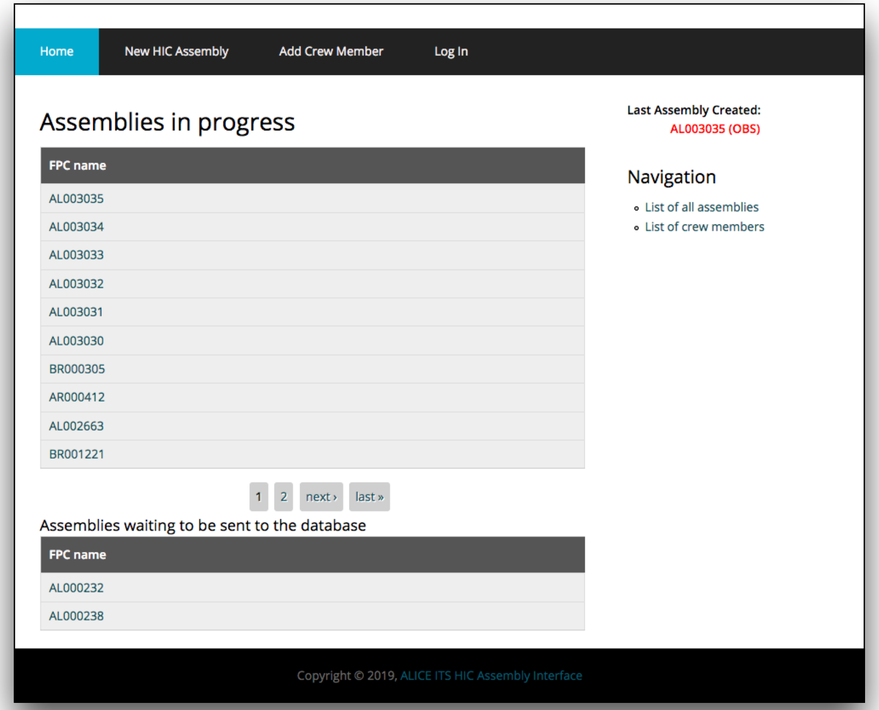
2. Chips are categorized on-the-fly as "SILVER" or "BRONZE" according to cuts applied to 47 parameters (Dead pixels, Fake Hit Rate, BackBias Max Voltage, ...).

3. The tray map shows "OK/NK" for each chip according to the selected category.

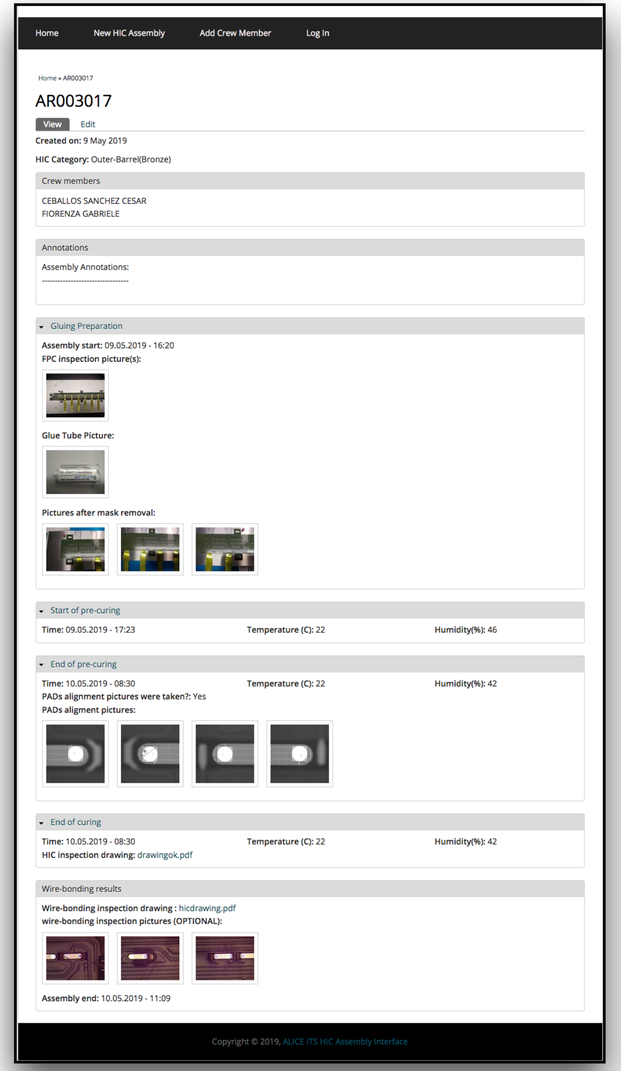
Outer Barrel HIC assembly - HIC Assembly Interface

MPD - ITS

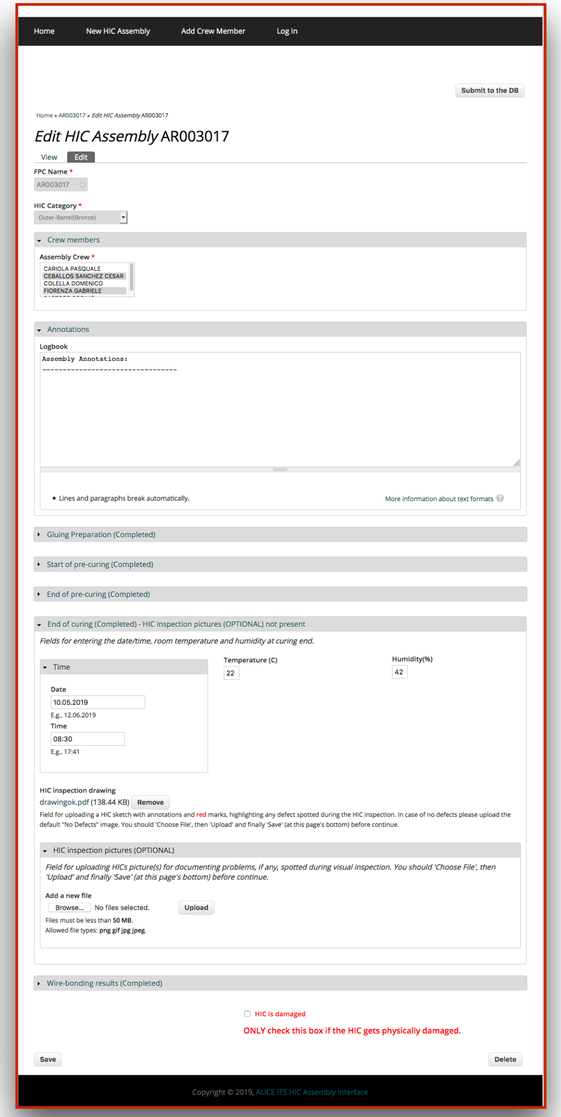
HIC Assembly Interface



View mode



Edit mode



It allows to:

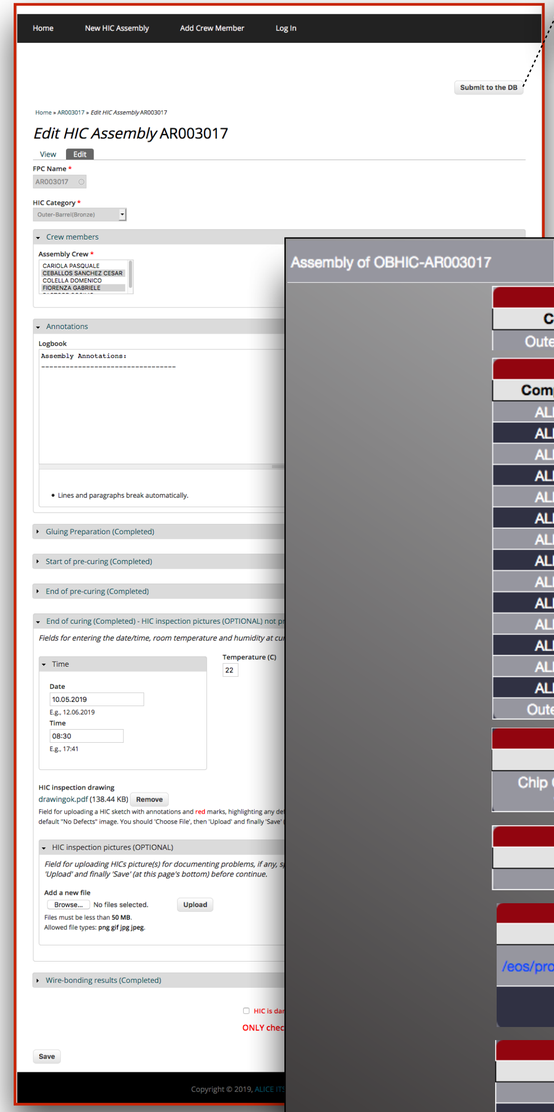
- Collect the assembly information.
- Keep track of all assemblies on site at different stages.
- Register the new HIC component and assembly activity into the project's DB and to send the assembly data to EOS (offline).

Outer Barrel HIC assembly - HIC Assembly Interface

MPD - ITS

All data have been collected

Submit to the DB



New HIC Component

Component ID	Component Type	Supplier Component ID	Description	Lot ID	Package ID	Phys. Status	Func. Status
OBHIC-AR003017	Outer Barrel HIC Module	-	-	-	-	OK	OBHIC-BRONZE

Parents

Component Type	Component ID	Position in this Component	Description
None			

Children

Component Type	Component ID	Position	Description
ALPIDEB Chip	T847782W11R26	HA1	
ALPIDEB Chip	T847782W11R40	HA2	
ALPIDEB Chip	T847782W11R03	HA3	
ALPIDEB Chip	T854192W11R14	HA4	
ALPIDEB Chip	T854192W11R03	HA5	
ALPIDEB Chip	T854192W11R08	HA6	
ALPIDEB Chip	T847782W11R30	HA7	
ALPIDEB Chip	T847782W11R36	HB1	
ALPIDEB Chip	T854192W11R00	HB2	
ALPIDEB Chip	T854186W19R45	HB3	
ALPIDEB Chip	T854186W19R38	HB4	
ALPIDEB Chip	T854186W19R27	HB5	
ALPIDEB Chip	T854186W19R33	HB6	
ALPIDEB Chip	T854186W19R19	HB7	
Outer Barrel FPC	AR003017	N/A	Outer Barrel FPC

New HIC Assembly Activity

Assembly of OBHIC-AR003017	OB HIC Assembly	09.05.2019	10.05.2019	Sezione INFN, Bari	Closed	OBHIC BRONZE Assembled

Output Components

Component Type	Component ID	Functional Status	Physical Status
Outer Barrel HIC Module	OBHIC-AR003017	unknown	OK

Input Components

Component Type	Component ID	Functional Status	Physical Status
ALPIDEB Chip	T854186W19R45	OK	OK
ALPIDEB Chip	T854186W19R33	OK	OK
ALPIDEB Chip	T847782W11R26	OK	OK
ALPIDEB Chip	T847782W11R40	OK	OK
ALPIDEB Chip	T854192W11R14	OK	OK
ALPIDEB Chip	T847782W11R30	OK	OK
ALPIDEB Chip	T854192W11R08	OK	OK
ALPIDEB Chip	T847782W11R03	OK	OK
ALPIDEB Chip	T854186W19R38	OK	OK
ALPIDEB Chip	T854186W19R19	OK	OK
ALPIDEB Chip	T847782W11R36	OK	OK
ALPIDEB Chip	T854192W11R03	OK	OK
ALPIDEB Chip	T854192W11R00	OK	OK
ALPIDEB Chip	T854186W19R27	OK	OK
Outer Barrel FPC	AR003017	OK	OK

Parameters

Name	Description	Value	Unit
Chip Classification Version	The version of the classification used by the ALPIDE-mass-testing-software	45	N/A

Attachments

Attachment Type	File
HIC Assembly Data	Results.html

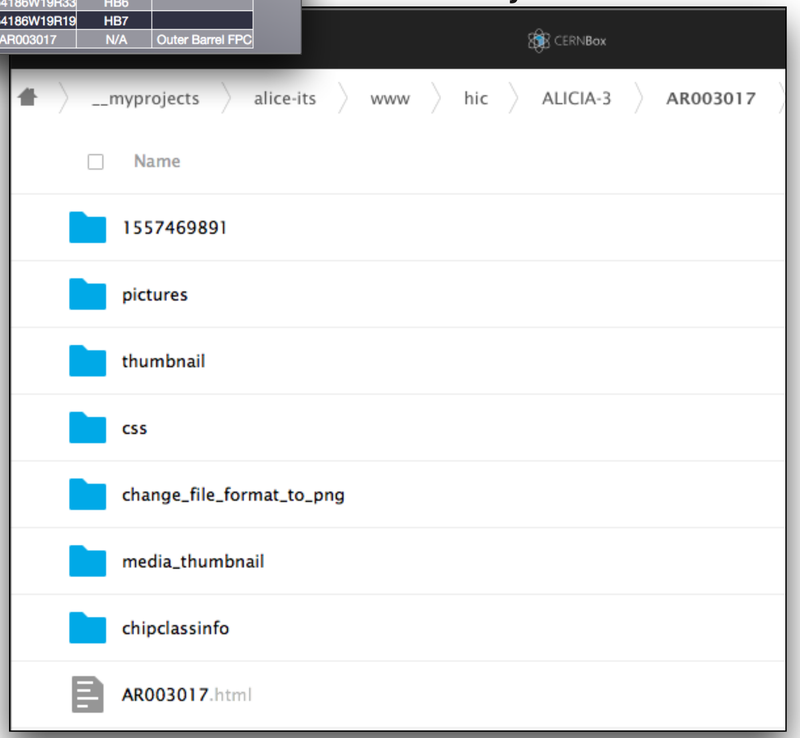
Uri

URI	Description
/eos/project/a/alice-its/hic/ALICIA-3/AR003017/chipclassinfo.tar.gz	path to chip classification info on EOS
http://cern.ch/hicassembly/ALICIA-3/AR003017/chipclassinfo.tar.gz	link to chip classification info on EOS

Members

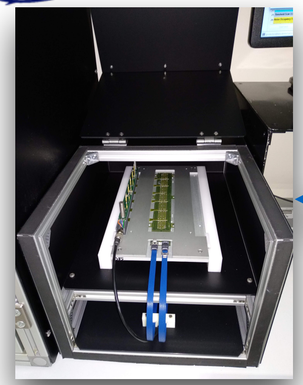
Name
CEBALLOS SANCHEZ CESAR
FIORENZA GABRIELE

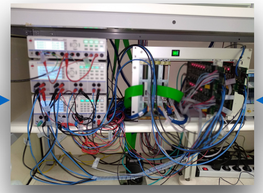
Assembly data on EOS

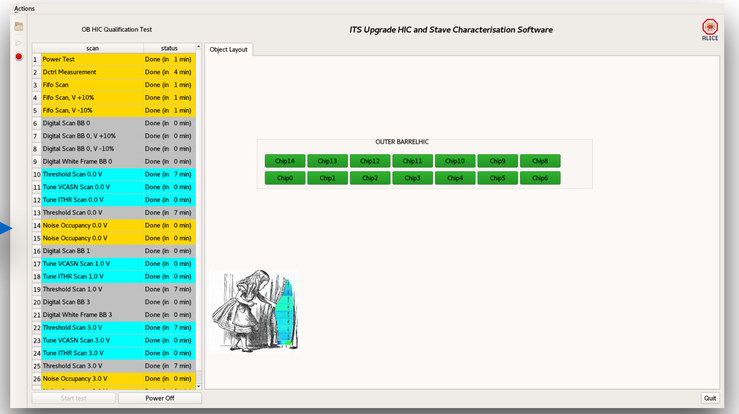


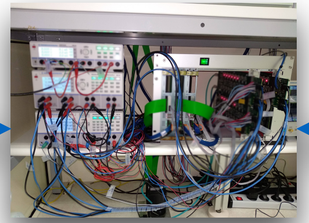
Similar procedure for testing at all levels

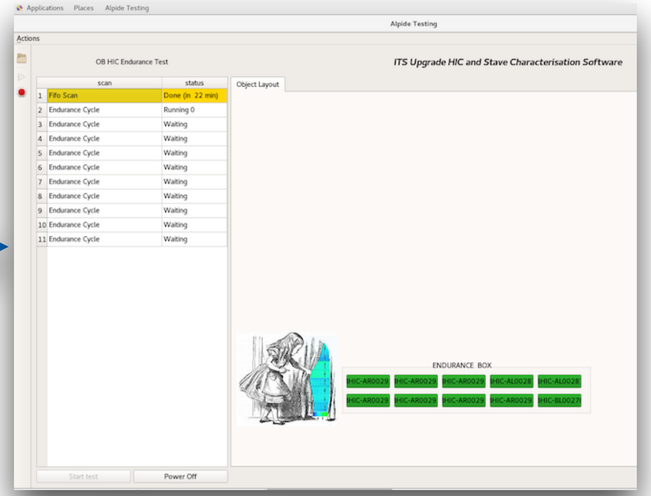
MPD - ITS

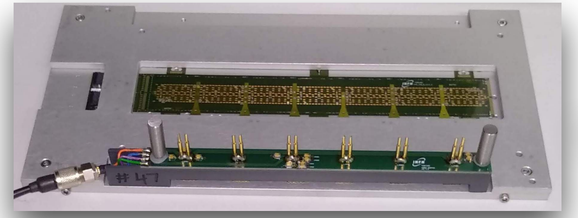










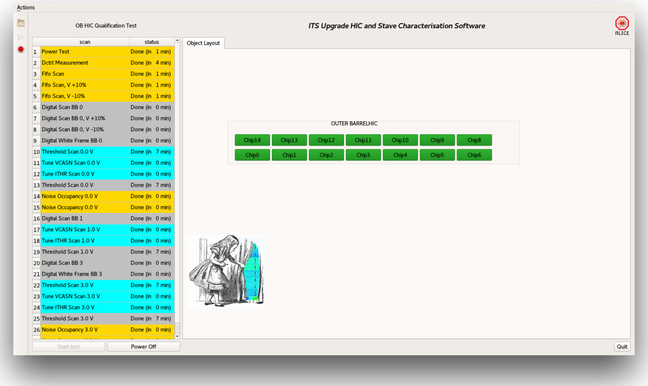


Thank you for your attention

Backup

Outer Barrel HIC assembly - Qualification test

Test



Test scans

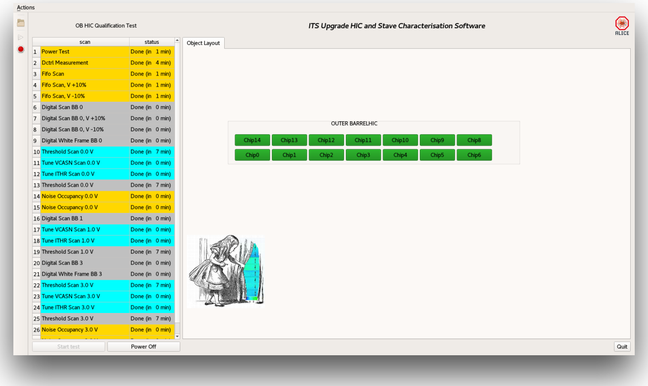
- Power Test.
- DCTRL Measurement.
- Fifo scan.
- Digital scan.
- Digital white frame.
- Threshold scan at 0V
- Threshold scan at -3V
- Threshold tuning at 0V
- Threshold tuning at -3V
- Noise occupancy at 0V
- Noise occupancy at -3V

Scans parameters for HIC classification

'Number of Working Chips',
'IDDA clocked',
'IDDD clocked',
'Maximum bias voltage',
'Back bias current 3V',
'FIFO errors (nominal)',
'FIFO errors (lower)',
'FIFO errors (upper)',
'FIFO exceptions (nominal)',
'FIFO exceptions (lower)',
'FIFO exceptions (upper)',
'Timeouts digital (nominal)',
'Timeouts digital (lower)',
'Timeouts digital (upper)',
'Corrupt events digital (nominal)',
'Corrupt events digital (lower)',
'Corrupt events digital (upper)',
'Bad pixels digital (nominal)',
'Bad pixels digital, worst chip (nominal)',
'Bad pixels digital (lower)',
'Bad pixels digital, worst chip (lower)',
'Bad pixels digital (upper)',
'Bad pixels digital, worst chip (upper)',
'Timeouts digital (nominal) BB 3V',
'Corrupt events digital (nominal) BB 3V',
'Bad pixels digital (nominal) BB 3V',
'Bad pixels digital, worst chip (nominal) BB 3V',
'Pixels without threshold tuned 0V',
'Pixels without thresh, worst, threshold tuned 0V',
'Pixels without threshold tuned 3V',
'Pixels without thresh, worst, threshold tuned 3V',
'Dead pixels threshold tuned 0V',
'Dead pixels, worst chip, threshold tuned 0V',
'Dead pixels threshold tuned 3V',
'Dead pixels, worst chip, threshold tuned 3V',
'Average noise threshold tuned 0V',
'Average noise threshold tuned 3V',
isy pixels 0V',
'Noisy pixels 3V',
'Noisy pixels masked 0V',
'Noisy pixels masked 3V',
'DCTRL worst max amplitude',
'DCTRL worst slope',
'DCTRL worst chi square',
'DCTRL worst rise time',
'DCTRL worst fall time',

Outer Barrel HIC assembly - Qualification test

Test



Test scans

- Power Test.
- DCTRL Measurement.
- Fifo scan. (& Fifo scan ± 10%)
- Digital scan.
- Digital white frame.
- Threshold scan at 0V
- Threshold scan at -3V
- Threshold tuning at 0V
- Threshold tuning at -3V
- Noise occupancy at 0V
- Noise occupancy at -3V

Scans parameters for HIC classification

'Number of Working Chips',
'IDDA clocked',
'IDDD clocked',
'Maximum bias voltage',
'Back bias current 3V',
'FIFO errors (nominal)',
'FIFO errors (lower)',
'FIFO errors (upper)',
'FIFO exceptions (nominal)',
'FIFO exceptions (lower)',
'FIFO exceptions (upper)',
'Timeouts digital (nominal)',
'Timeouts digital (lower)',
'Timeouts digital (upper)',
'Corrupt events digital (nominal)',
'Corrupt events digital (lower)',
'Corrupt events digital (upper)',
'Bad pixels digital (nominal)',
'Bad pixels digital, worst chip (nominal)',
'Bad pixels digital (lower)',
'Bad pixels digital, worst chip (lower)',
'Bad pixels digital (upper)',
'Bad pixels digital, worst chip (upper)',
'Timeouts digital (nominal) BB 3V',
'Corrupt events digital (nominal) BB 3V',
'Bad pixels digital (nominal) BB 3V',
'Bad pixels digital, worst chip (nominal) BB 3V',
'Pixels without threshold tuned 0V',
'Pixels without thresh, worst, threshold tuned 0V',
'Pixels without threshold tuned 3V',
'Pixels without thresh, worst, threshold tuned 3V',
'Dead pixels threshold tuned 0V',
'Dead pixels, worst chip, threshold tuned 0V',
'Dead pixels threshold tuned 3V',
'Dead pixels, worst chip, threshold tuned 3V',
'Average noise threshold tuned 0V',
'Average noise threshold tuned 3V',
'Noisy pixels 0V',
'Noisy pixels 3V',
'Noisy pixels masked 0V',
'Noisy pixels masked 3V',
'DCTRL worst max amplitude',
'DCTRL worst slope',
'DCTRL worst chi square',
'DCTRL worst rise time',
'DCTRL worst fall time',

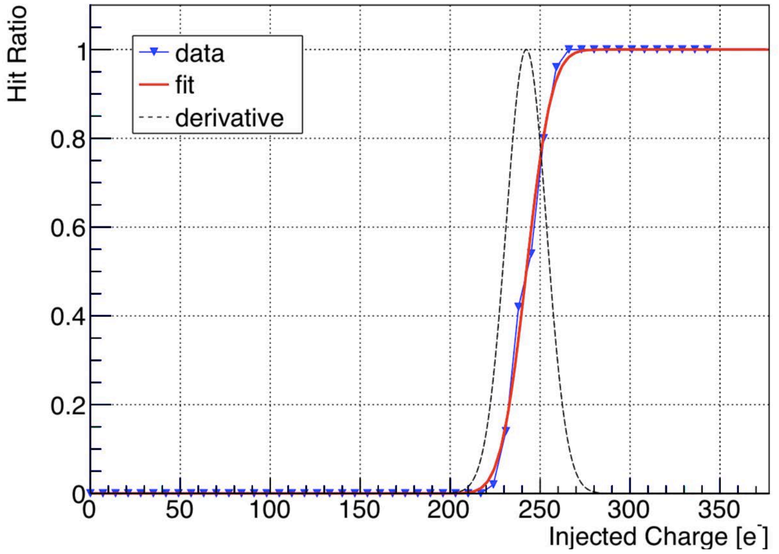
Outer Barrel HIC assembly - Qualification test

Treshold Scan:

» It determines the threshold and noise of each pixel in units of electrons.

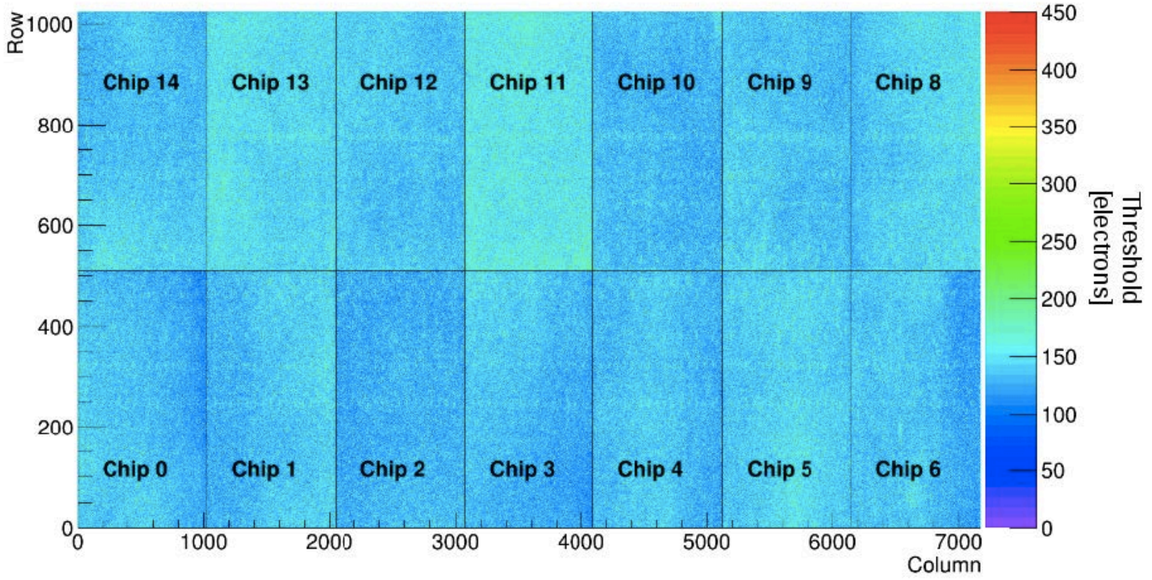
» Checks out the analog performance of the chip.

» Chip by chip threshold tuning



Parameter cuts for HIC classification. (Gold otherwise)

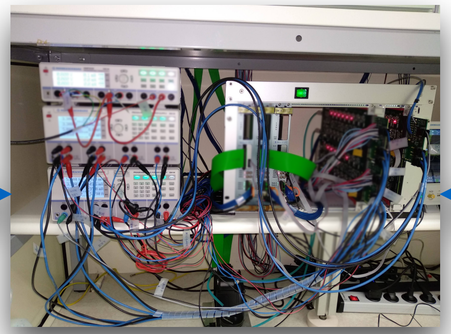
Parameter	SILVER	BRONZE	NOT WORKING
Dead pixels, worst chip, threshold tuned OV	> 50	> 2100	> 5243
Pixels without thresh, worst, threshold tuned OV	> 5243	> 26214	> 52429
Average noise threshold tuned OV	> 10 el	-	-

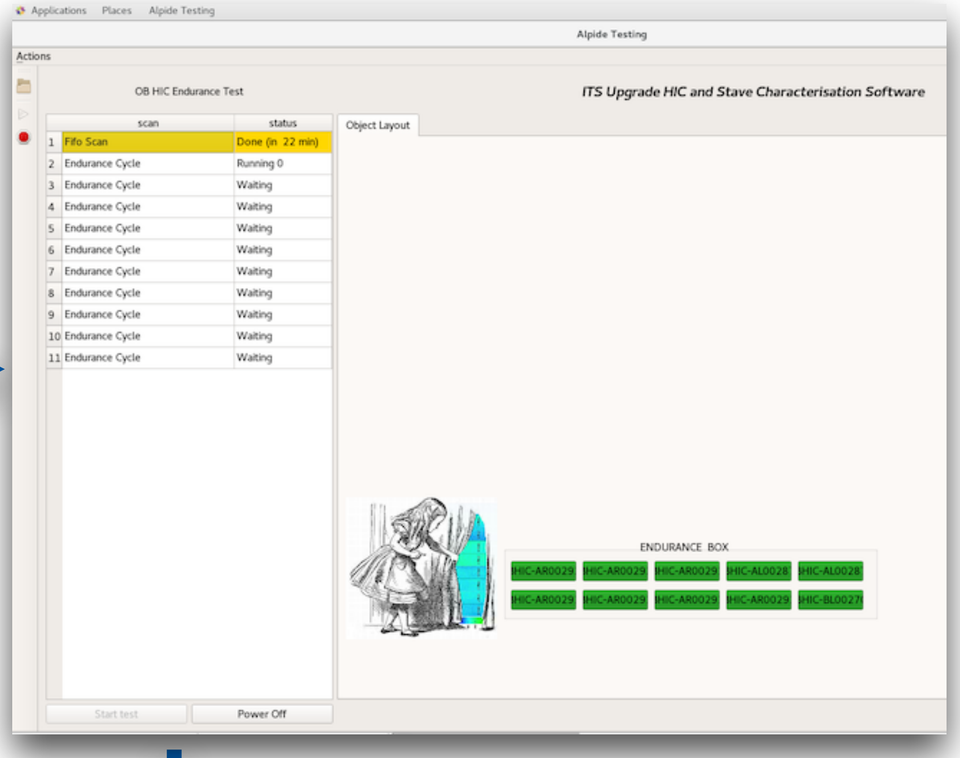


- At the end of each **scan**, the cuts on the related parameters are applied and the scan gets the classification of the **worst-ranking parameter**.
- At the end of the test, the **HIC** is classified as the **worst-ranking scan**.

Outer Barrel HIC assembly - Endurance test

A series of power cycles and FIFO scans (writing and reading back registers) for testing the chips control interface.





Test characteristics

- Number of slices: 10
- Cycles per slice: 13
- Slice time limit: 8h
- Uptime: 1600s
- Down time: 200s
- Number of triggers: 100000

Impedance, Qualification and **Endurance** tests are performed on **all** HICs.

The API functions: Ex. ‘Activity Create’

AlucmsWebAPI

Click here for a complete list of operations.

ActivityCreate

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
activityTypeID:	
locationID:	
lotID:	
activityName:	
startDate:	
endDate:	
position:	
resultID:	
statusID:	
userID:	

Invoke

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /AlucmswebAPI.asmx HTTP/1.1
Host: alucmsapi.web.cern.ch
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/ActivityCreate"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```

The API functions: Ex. ‘Activity Create’

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ActivityCreate xmlns="http://tempuri.org/">
      <activityTypeID>int</activityTypeID>
      <locationID>int</locationID>
      <lotID>string</lotID>
      <activityName>string</activityName>
      <startDate>string</startDate>
      <endDate>string</endDate>
      <position>string</position>
      <resultID>int</resultID>
      <statusID>int</statusID>
      <userID>int</userID>
    </ActivityCreate>
  </soap:Body>
</soap:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ActivityCreateResponse xmlns="http://tempuri.org/">
      <ActivityCreateResult>
        <ErrorCode>int</ErrorCode>
        <ErrorMessage>string</ErrorMessage>
        <ID>int</ID>
      </ActivityCreateResult>
    </ActivityCreateResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /AlucmswebAPI.asmx HTTP/1.1
Host: alucmsapi.web.cern.ch
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <ActivityCreate xmlns="http://tempuri.org/">
      <activityTypeID>int</activityTypeID>
      <locationID>int</locationID>
      <lotID>string</lotID>
      <activityName>string</activityName>
      <startDate>string</startDate>
```

The API functions: Ex. ‘Activity Create’

```
      <startDate>string</startDate>
      <endDate>string</endDate>
      <position>string</position>
      <resultID>int</resultID>
      <statusID>int</statusID>
      <userID>int</userID>
    </ActivityCreate>
  </soap12:Body>
</soap12:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <ActivityCreateResponse xmlns="http://tempuri.org/">
      <ActivityCreateResult>
        <ErrorCode>int</ErrorCode>
        <ErrorMessage>string</ErrorMessage>
        <ID>int</ID>
      </ActivityCreateResult>
    </ActivityCreateResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP GET

The following is a sample HTTP GET request and response. The placeholders shown need to be replaced with actual values.

```
GET /AlucmswebAPI.asmx/ActivityCreate?activityTypeID=string&locationID=string&lotID=string&activityName=string&startDate=string&endDate=string&position=string&resultID=string&statusID=s
Host: alucmsapi.web.cern.ch
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<FunctionResult xmlns="http://tempuri.org/">
  <ErrorCode>int</ErrorCode>
  <ErrorMessage>string</ErrorMessage>
  <ID>int</ID>
</FunctionResult>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /AlucmswebAPI.asmx/ActivityCreate HTTP/1.1
Host: alucmsapi.web.cern.ch
Content-Type: application/x-www-form-urlencoded
```

The API functions: Ex. ‘Activity Create’

```
      <ActivityCreateResult>
        <ErrorCode>int</ErrorCode>
        <ErrorMessage>string</ErrorMessage>
        <ID>int</ID>
      </ActivityCreateResult>
    </ActivityCreateResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP GET

The following is a sample HTTP GET request and response. The placeholders shown need to be replaced with actual values.

```
GET /AlucmswebAPI.asmx/ActivityCreate?activityTypeID=string&locationID=string&lotID=string&activityName=string&startDate=string&endDate=string&position=string&resultID=string&statusID=s
Host: alucmsapi.web.cern.ch
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<FunctionResult xmlns="http://tempuri.org/">
  <ErrorCode>int</ErrorCode>
  <ErrorMessage>string</ErrorMessage>
  <ID>int</ID>
</FunctionResult>
```

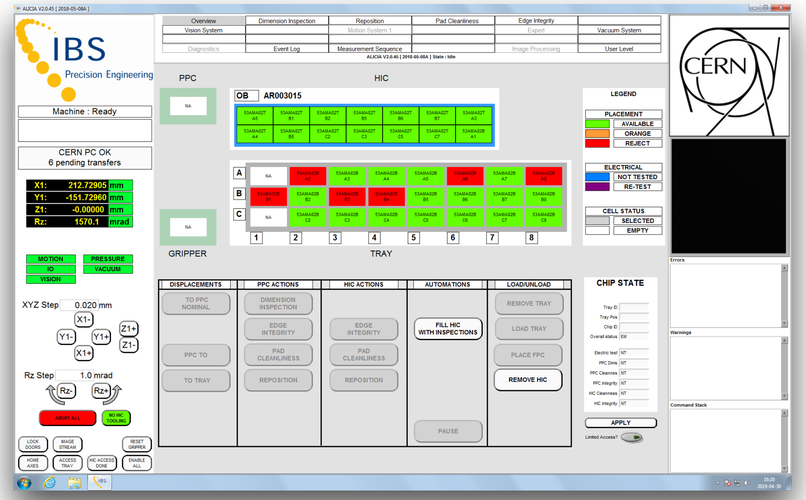
HTTP POST

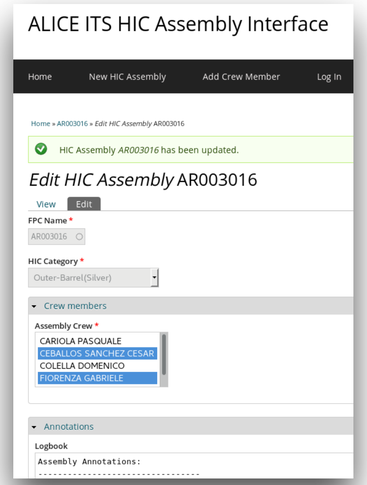
The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

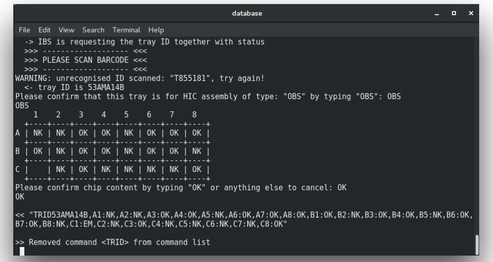
```
POST /AlucmswebAPI.asmx/ActivityCreate HTTP/1.1
Host: alucmsapi.web.cern.ch
Content-Type: application/x-www-form-urlencoded
Content-Length: length

activityTypeID=string&locationID=string&lotID=string&activityName=string&startDate=string&endDate=string&position=string&resultID=string&statusID=string&userID=string
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<FunctionResult xmlns="http://tempuri.org/">
  <ErrorCode>int</ErrorCode>
  <ErrorMessage>string</ErrorMessage>
  <ID>int</ID>
</FunctionResult>
```

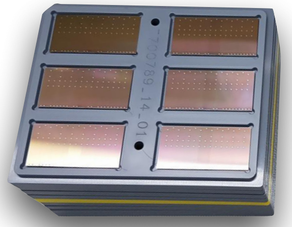
OBHIC Assembly - Using the API functions





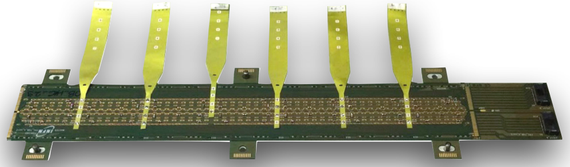


2 x 7 Chips



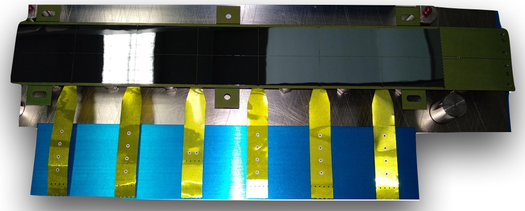
+

1 FPC



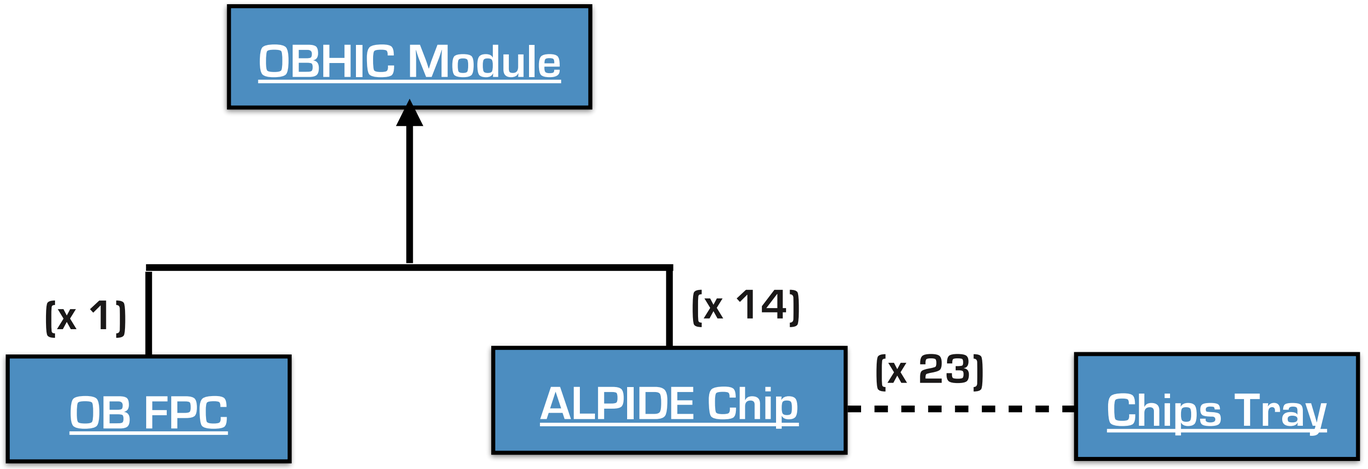
=

1 OBHIC



OBHIC Assembly - Using the API functions

1. Defining the **Component Type** (Web Interface)

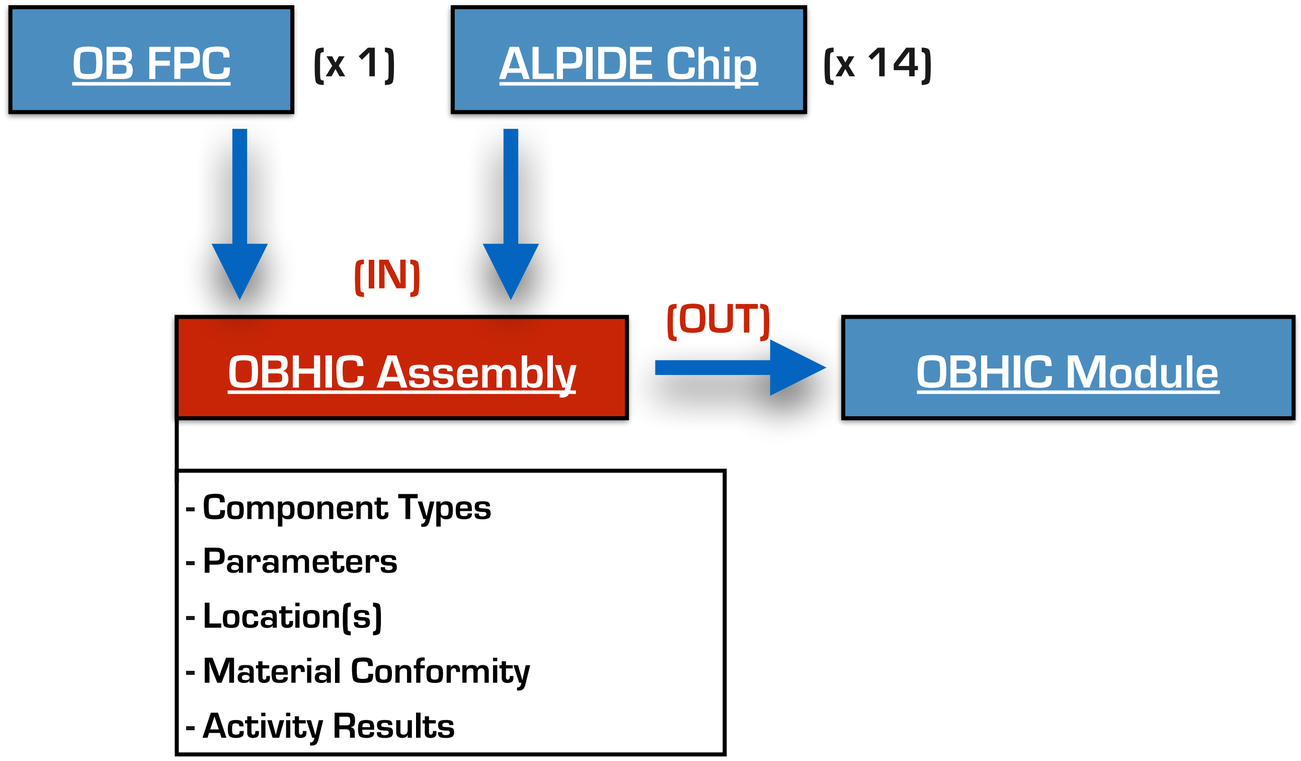


Component Categories

- Pixel Chip
- Chip Container
- Flexible Printed Circuit
- HIC Module

OBHIC Assembly - Using the API functions

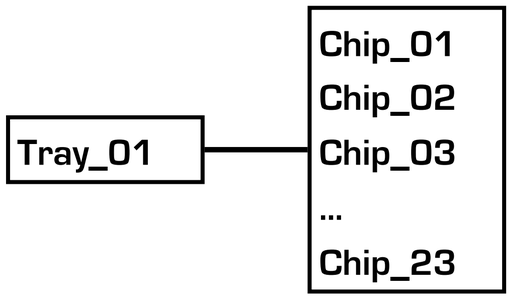
2. Defining the **Activity Type** (Web Interface)



OBHIC Assembly - Using the API functions

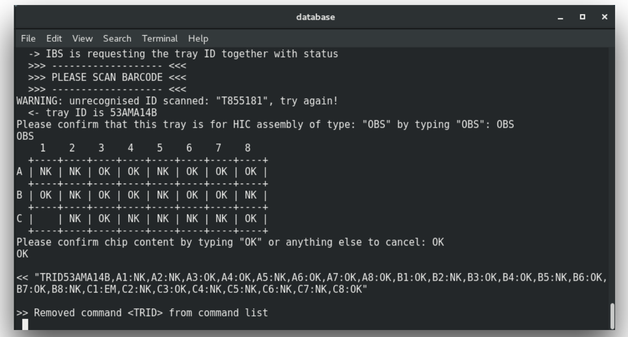
3. Creating the **specific OBHIC Component**

(Components already existing on the DB)



OBHIC Assembly - Using the API functions

3. Creating the **specific OBHIC Component**

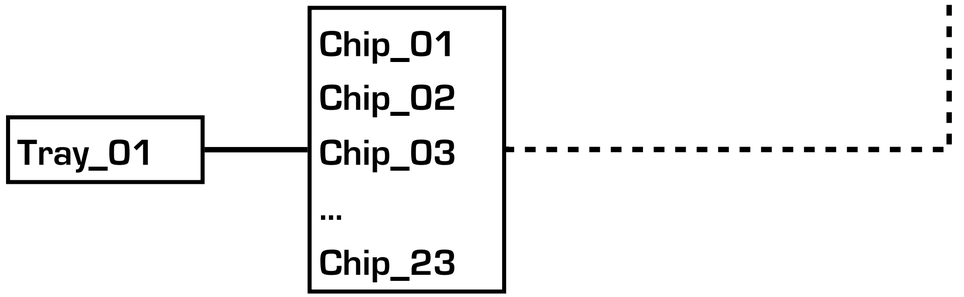


1. <u>Scan tray's barcode and load the tray into the MAM</u>

The following information is retrieved:

- The **Component ID** of all chips belonging to that tray.
- The **Position of each chip** in the tray.
- The **Physical Status and Functional Status** of each chip.
- The tray map shows **"OK/NK"** for each chip

The chip alignment starts

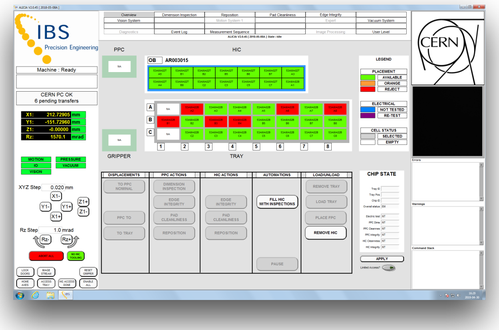


FPC_01

OBHIC Assembly - Using the API functions

3. Creating the **specific OBHIC Component**

2. ‘Place FPC’ on the MAM (Interface and Physically)



Tray_01

- Chip_01
- Chip_02
- Chip_03
- ...
- Chip_23

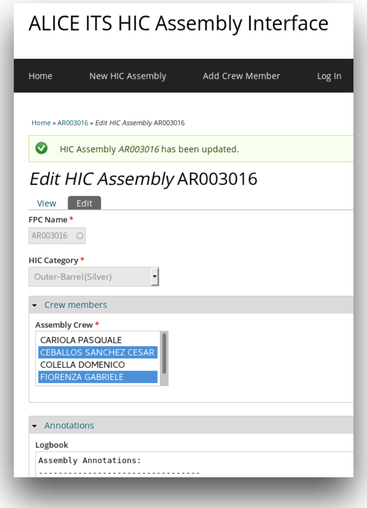
FPC_01

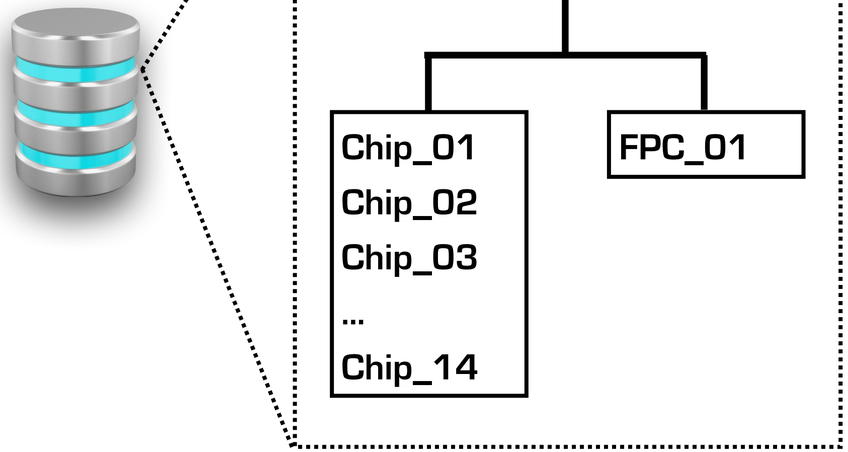
OBHIC Assembly - Using the API functions

3. Creating the **specific OBHIC Component**

3. <u>Submit the assembly to the DB</u>

A **new component** of type OB HIC Module is created in the DB with a unique Component ID, composed by the **specific FPC and Chips** used in the assembly process.

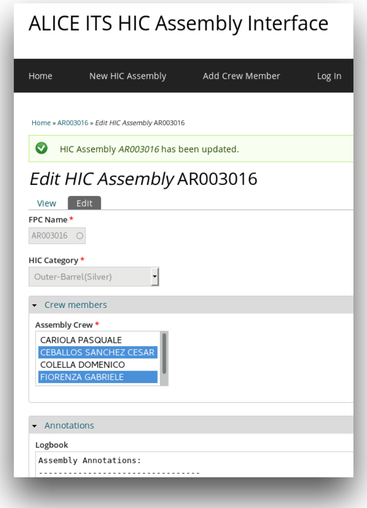


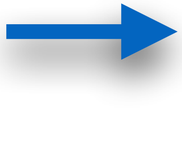


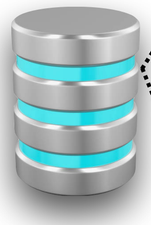
OBHIC Assembly - Using the API functions

4. Creating the **specific OBHIC Assembly Activity**

A new activity of type OB HIC Assembly will be created on the DB (with a unique Activity Name) using as **Output Component** the OB HIC created on step 3 and as **Input Components** those already related to it in the same step 3.







Assembly of OBHIC_01

- **Output Component**
- **Input Components**
- **Location**
- **Parameters Value**
- **Attachments**
- **Result**
- **Members**

And then the OBHIC tests begin...