

# 【BSMT-R4P4】 APIリファレンス

Version: v1.5

Rev	更新日	更新内容・理由
v1.3	2023.11.27	初版発行
v1.4	2024.2.19	スケジュール機能の追加
v1.5	2025.1.30	バージョンを揃える

## 目次

### 1. クエリ (Query)

- 1-1. events
- 1-2. notificationEvents
- 1-3. eventTriggers
- 1-4. writeRemoteOutportEvents
- 1-5. inports
- 1-6. logEvents
- 1-7. logInports
- 1-8. logOutports
- 1-9. notificationTargets
- 1-10. outports
- 1-11. users
- 1-12. cloud
- 1-13. dateTime
- 1-14. deviceInfo
- 1-15. eventTimelines
- 1-16. network
- 1-17. latestFirmware
- 1-18. verifyUser

### 2. ミューテーション (Mutation)

- 2-1. updateCloud
- 2-2. updateDateTime
- 2-3. createEvent
- 2-4. updateEvent
- 2-5. deleteEvent
- 2-6. testEvent
- 2-7. createEventTrigger
- 2-8. updateEventTrigger
- 2-9. deleteEventTrigger
- 2-10. updateInport
- 2-11. updateNetwork
- 2-12. createNotificationTarget
- 2-13. updateNotificationTarget
- 2-14. deleteNotificationTarget
- 2-15. notify
- 2-16. updateOutport
- 2-17. writeOutport
- 2-18. onlineFirmwareUpdate
- 2-19. reboot
- 2-20. factoryReset
- 2-21. createUser
- 2-22. updateUser
- 2-23. deleteUser

### 3. 型定義

- 3-1. オブジェクト型
  - 3-1-1. Cloud

3-1-2. CloudAzureConfig  
3-1-3. CloudSmartioConfig  
3-1-4. DateTime  
3-1-5. DateTimeConfig  
3-1-6. DeviceInfo  
3-1-7. DeviceFirmware  
3-1-8. UpTime  
3-1-9. Event  
3-1-10. EventConnection  
3-1-11. EventEdge  
3-1-12. EventNotification  
3-1-13. EventNotificationConnection  
3-1-14. EventNotificationEdge  
3-1-15. EventTrigger  
3-1-16. EventTriggerConnection  
3-1-17. EventTriggerEdge  
3-1-18. EventTriggerInport  
3-1-19. EventTriggerOutport  
3-1-20. EventTriggerSchedule  
3-1-21. EventWriteRemoteOutport  
3-1-22. EventWriteRemoteOutportConnection  
3-1-23. EventWriteRemoteOutportEdge  
3-1-24. Inport  
3-1-25. InportConnection  
3-1-26. InportEdge  
3-1-27. LogEvent  
3-1-28. LogEventConnection  
3-1-29. LogEventEdge  
3-1-30. LogInport  
3-1-31. LogInportConnection  
3-1-32. LogInportEdge  
3-1-33. LogOutport  
3-1-34. LogOutportConnection  
3-1-35. LogOutportEdge  
3-1-36. NotificationTarget  
3-1-37. NotificationTargetConnection  
3-1-38. NotificationTargetEdge  
3-1-39. NotificationTargetEmail  
3-1-40. NotificationTargetWebhook  
3-1-41. Outport  
3-1-42. OutportConnection  
3-1-43. OutportEdge  
3-1-44. PageInfo  
3-1-45. RemoteOutport  
3-1-46. User  
3-1-47. UserConnection  
3-1-48. UserEdge  
3-1-49. EventTimelinesResponse

- 3-1-50. EventTimeline
- 3-1-51. ScheduleEventTriggerDate
- 3-1-52. ScheduleEventTriggerDateMonths
- 3-1-53. ScheduleEventTriggerDateDayOfWeeks
- 3-1-54. ScheduleEventTriggerTime
- 3-1-55. ScheduleEventTriggerTimeRepeat
- 3-1-56. Network
- 3-1-57. NetworkConfigDhcp
- 3-1-58. NetworkConfigStatic
- 3-1-59. LatestFirmware
- 3-1-60. OnlineFirmwareUpdateResponse
- 3-1-61. RebootResponse
- 3-1-62. FactoryResetResponse
- 3-2. インターフェース型
  - 3-2-1. Node
- 3-3. 入力型
  - 3-3-1. UpdateCloudInput
  - 3-3-2. OneofCloudConfigInput
  - 3-3-3. CloudConfigAzureInput
  - 3-3-4. CloudConfigSmartIOLInput
  - 3-3-5. CloudConfigNoneInput
  - 3-3-6. UpdateDateTimeInput
  - 3-3-7. DateTimeConfigInput
  - 3-3-8. EventNotificationOrder
  - 3-3-9. EventNotificationWhereInput
  - 3-3-10. EventOrder
  - 3-3-11. EventTriggerInportOrder
  - 3-3-12. EventTriggerInportWhereInput
  - 3-3-13. EventTriggerOrder
  - 3-3-14. EventTriggerOutportOrder
  - 3-3-15. EventTriggerOutportWhereInput
  - 3-3-16. EventTriggerScheduleOrder
  - 3-3-17. EventTriggerScheduleWhereInput
  - 3-3-18. EventTriggerWhereInput
  - 3-3-19. EventWhereInput
  - 3-3-20. EventWriteRemoteOutportOrder
  - 3-3-21. EventWriteRemoteOutportWhereInput
  - 3-3-22. InportOrder
  - 3-3-23. InportWhereInput
  - 3-3-24. LogEventOrder
  - 3-3-25. LogEventWhereInput
  - 3-3-26. LogInportOrder
  - 3-3-27. LogInportWhereInput
  - 3-3-28. LogOutportOrder
  - 3-3-29. LogOutportWhereInput
  - 3-3-30. NotificationTargetEmailOrder
  - 3-3-31. NotificationTargetEmailWhereInput
  - 3-3-32. NotificationTargetOrder

3-3-33. NotificationTargetWebhookOrder  
3-3-34. NotificationTargetWebhookWhereInput  
3-3-35. NotificationTargetWhereInput  
3-3-36. OutportOrder  
3-3-37. OutportWhereInput  
3-3-38. RemoteOutportOrder  
3-3-39. RemoteOutportWhereInput  
3-3-40. UserOrder  
3-3-41. UserWhereInput  
3-3-42. CreateEventInput  
3-3-43. UpdateEventInput  
3-3-44. CreateNotificationEventInput  
3-3-45. CreateWriteRemoteOutportEventInput  
3-3-46. UpdateWriteRemoteOutportEventInput  
3-3-47. CreateRemoteOutportInput  
3-3-48. UpdateRemoteOutportInput  
3-3-49. CreateEventTriggerInput  
3-3-50. UpdateEventTriggerInput  
3-3-51. OneofEventTrigger  
3-3-52. InportEventTriggerInput  
3-3-53. OneofInportEventTriggerCondition  
3-3-54. InportEventTriggerConditionAll  
3-3-55. InportEventTriggerConditionAny  
3-3-56. OutportEventTriggerInput  
3-3-57. OneofOutportEventTriggerCondition  
3-3-58. OutportEventTriggerConditionAll  
3-3-59. OutportEventTriggerConditionAny  
3-3-60. ScheduleEventTriggerInput  
3-3-61. ScheduleEventTriggerDateInput  
3-3-62. ScheduleEventTriggerDateMonthsInput  
3-3-63. ScheduleEventTriggerDateDayOfWeeksInput  
3-3-64. ScheduleEventTriggerTimeInput  
3-3-65. ScheduleEventTriggerTimeRepeatInput  
3-3-66. UpdateInportInput  
3-3-67. UpdateNetworkInput  
3-3-68. OneofNetworkConfigInput  
3-3-69. NetworkConfigDhcpInput  
3-3-70. NetworkConfigStaticInput  
3-3-71. CreateNotificationTargetInput  
3-3-72. UpdateNotificationTargetInput  
3-3-73. OneofCreateNotificationTarget  
3-3-74. CreateNotificationTargetEmailInput  
3-3-75. OneofSMTPAuthInput  
3-3-76. SMTPAuthNoneInput  
3-3-77. SMTPAuthPlaneInput  
3-3-78. CreateNotificationTargetWebhookInput  
3-3-79. NotifyInput  
3-3-80. UpdateOutportInput

- 3-3-81. WriteOutputInput
- 3-3-82. OneofWriteOutputOperation
- 3-3-83. WriteOutputOperationOn
- 3-3-84. WriteOutputOperationOff
- 3-3-85. OnlineFirmwareUpdateInput
- 3-3-86. VerifyUserInput
- 3-3-87. CreateUserInput
- 3-3-88. UpdateUserInput

#### 3-4. 列挙型

- 3-4-1. CloudKindEnum
- 3-4-2. CloudStateEnum
- 3-4-3. EventNotificationOrderField
- 3-4-4. EventOrderField
- 3-4-5. EventTriggerInportOrderField
- 3-4-6. EventTriggerKindEnum
- 3-4-7. EventTriggerOrderField
- 3-4-8. EventTriggerOutputOrderField
- 3-4-9. EventTriggerScheduleOrderField
- 3-4-10. EventWriteRemoteOutputOrderField
- 3-4-11. InportEventTriggerConditionEnum
- 3-4-12. InportEventTriggerInportStateEnum
- 3-4-13. InportOrderField
- 3-4-14. InportPortNoEnum
- 3-4-15. LogEventOrderField
- 3-4-16. LogInportOrderField
- 3-4-17. LogOutputOrderField
- 3-4-18. NotificationTargetEmailOrderField
- 3-4-19. NotificationTargetKindEnum
- 3-4-20. NotificationTargetOrderField
- 3-4-21. NotificationTargetWebhookOrderField
- 3-4-22. OrderDirection
- 3-4-23. OutputEventTriggerConditionEnum
- 3-4-24. OutputEventTriggerOutputStateEnum
- 3-4-25. OutputInitialStateOnStartupEnum
- 3-4-26. OutputOrderField
- 3-4-27. OutputPortNoEnum
- 3-4-28. RemoteOutputOperationEnum
- 3-4-29. RemoteOutputOrderField
- 3-4-30. SMTPAuthEnum
- 3-4-31. SMTPEncryptionEnum
- 3-4-32. ScheduleEventTriggerDateKindEnum
- 3-4-33. UserKindEnum
- 3-4-34. UserOrderField
- 3-4-35. InportStateEnum
- 3-4-36. OutputStateEnum

#### 3-5. スカラー型

- 3-5-1. ID
- 3-5-2. Time

3-5-3. Map

3-5-4. Dummy

3-5-5. String

3-5-6. Int

3-5-7. Cursor

3-5-8. Boolean

# 1. クエリ (Query)

## 1-1. events

戻り値: EventConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first _n_ elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last _n_ elements from the list.
orderBy	<u>EventOrder</u>	Ordering options for Events returned from the connection.
where	<u>EventWhereInput</u>	Filtering options for Events returned from the connection.

使用例:

```
query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: EventOrder,
  $where: EventWhereInput
){
  events(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}
```

## 1-2. notificationEvents

戻り値: EventNotificationConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first _n_ elements from the list.



引数	型	説明
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last _n_ elements from the list.
orderBy	<u>EventNotificationOrder</u>	Ordering options for EventNotifications returned from the connection.
where	<u>EventNotificationWhereInput</u>	Filtering options for EventNotifications returned from the connection.

使用例:

```
query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: EventNotificationOrder,
  $where: EventNotificationWhereInput
){
  notificationEvents(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}
```

### 1-3. eventTriggers

戻り値: EventTriggerConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first _n_ elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last _n_ elements from the list.
orderBy	<u>EventTriggerOrder</u>	Ordering options for EventTriggers returned from the connection.
where	<u>EventTriggerWhereInput</u>	Filtering options for EventTriggers returned from the connection.

使用例:

```
query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
```

```

$last: Int,
$orderBy: EventTriggerOrder,
$where: EventTriggerWhereInput
){
  eventTriggers(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-4. writeRemoteOutportEvents

戻り値: EventWriteRemoteOutportConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <code>_n_</code> elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <code>_n_</code> elements from the list.
orderBy	<u>EventWriteRemoteOutportOrder</u>	Ordering options for EventWriteRemoteOutports returned from the connection.
where	<u>EventWriteRemoteOutportWhereInput</u>	Filtering options for EventWriteRemoteOutports returned from the connection.

使用例:

```

query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: EventWriteRemoteOutportOrder,
  $where: EventWriteRemoteOutportWhereInput
){
  writeRemoteOutportEvents(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-5. inports

戻り値: InportConnection!

引数:

引数	型	説明
after	<a href="#">Cursor</a>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <code>_n_</code> elements from the list.
before	<a href="#">Cursor</a>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <code>_n_</code> elements from the list.
orderBy	<a href="#">InportOrder</a>	Ordering options for Inports returned from the connection.
where	<a href="#">InportWhereInput</a>	Filtering options for Inports returned from the connection.

使用例:

```
query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: InportOrder,
  $where: InportWhereInput
){
  inports(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}
```

## 1-6. logEvents

戻り値: [LogEventConnection!](#)

引数:

引数	型	説明
after	<a href="#">Cursor</a>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <code>_n_</code> elements from the list.
before	<a href="#">Cursor</a>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <code>_n_</code> elements from the list.
orderBy	<a href="#">LogEventOrder</a>	Ordering options for LogEvents returned from the connection.
where	<a href="#">LogEventWhereInput</a>	Filtering options for LogEvents returned from the connection.

使用例:

```
query (
```

```

$after: Cursor,
$first: Int,
$before: Cursor,
$last: Int,
$orderBy: LogEventOrder,
$where: LogEventWhereInput
){
  logEvents(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-7. logInports

戻り値: LogInportConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <u>_n_</u> elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <u>_n_</u> elements from the list.
orderBy	<u>LogInportOrder</u>	Ordering options for LogInports returned from the connection.
where	<u>LogInportWhereInput</u>	Filtering options for LogInports returned from the connection.

使用例:

```

query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: LogInportOrder,
  $where: LogInportWhereInput
){
  logInports(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-8. logOutports

戻り値: LogOutportConnection!

引数:

引数	型	説明
after	<a href="#">Cursor</a>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <code>_n_</code> elements from the list.
before	<a href="#">Cursor</a>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <code>_n_</code> elements from the list.
orderBy	<a href="#">LogOutputportOrder</a>	Ordering options for LogOutputports returned from the connection.
where	<a href="#">LogOutputportWhereInput</a>	Filtering options for LogOutputports returned from the connection.

使用例:

```

query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: LogOutputportOrder,
  $where: LogOutputportWhereInput
){
  logOutputports(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-9. notificationTargets

戻り値: [NotificationTargetConnection](#) !

引数:

引数	型	説明
after	<a href="#">Cursor</a>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <code>_n_</code> elements from the list.
before	<a href="#">Cursor</a>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <code>_n_</code> elements from the list.
orderBy	<a href="#">NotificationTargetOrder</a>	Ordering options for NotificationTargets returned from the connection.
where	<a href="#">NotificationTargetWhereInput</a>	Filtering options for NotificationTargets returned from the connection.

使用例:

```

query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: NotificationTargetOrder,
  $where: NotificationTargetWhereInput
){
  notificationTargets(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-10. outports

戻り値: OutportConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first <u>_n_</u> elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last <u>_n_</u> elements from the list.
orderBy	<u>OutportOrder</u>	Ordering options for Outports returned from the connection.
where	<u>OutportWhereInput</u>	Filtering options for Outports returned from the connection.

使用例:

```

query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: OutportOrder,
  $where: OutportWhereInput
){
  outports(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}

```

## 1-11. users

戻り値: UserConnection!

引数:

引数	型	説明
after	<u>Cursor</u>	Returns the elements in the list that come after the specified cursor.
first	Int	Returns the first _n_ elements from the list.
before	<u>Cursor</u>	Returns the elements in the list that come before the specified cursor.
last	Int	Returns the last _n_ elements from the list.
orderBy	<u>UserOrder</u>	Ordering options for Users returned from the connection.
where	<u>UserWhereInput</u>	Filtering options for Users returned from the connection.

使用例:

```
query (
  $after: Cursor,
  $first: Int,
  $before: Cursor,
  $last: Int,
  $orderBy: UserOrder,
  $where: UserWhereInput
){
  users(after: $after, first: $first, before: $before, last: $last, orderBy: $orderBy, where: $where) {
    # フィールドを指定
  }
}
```

## 1-12. cloud

クラウドの接続状態及び、設定情報を取得します。

戻り値: Cloud!

使用例:

```
{
  cloud {
    # フィールドを指定
  }
}
```

## 1-13. dateTime

現在時刻及び、時刻設定を取得します。

戻り値: [DateTime!](#)

使用例:

```
{
  dateTime{
    # フィールドを指定
  }
}
```

## 1-14. deviceInfo

本機の情報を取得します。

戻り値: [DeviceInfo!](#)

使用例:

```
{
  deviceInfo{
    # フィールドを指定
  }
}
```

## 1-15. eventTimelines

戻り値: [EventTimelinesResponse!](#)

引数:

引数	型	説明
start	<u><a href="#">Time!</a></u>	
until	<u><a href="#">Time!</a></u>	
limitPerDay	Int	

使用例:

```
query (
  $start: Time!,
  $until: Time!,
  $limitPerDay: Int
){
  eventTimelines(start: $start, until: $until, limitPerDay: $limitPerDay) {
    # フィールドを指定
  }
}
```



## 1-16. network

ネットワークの状態及び、設定情報を取得します。

戻り値: Network!

使用例:

```
{
  network{
    # フィールドを指定
  }
}
```

## 1-17. latestFirmware

最新のファームウェア情報を取得します。

TOPレベルエラー: FIRMWARE\_INFORMATION\_FETCH\_FAILED

戻り値: LatestFirmware!

使用例:

```
{
  latestFirmware{
    # フィールドを指定
  }
}
```

## 1-18. verifyUser

ユーザの認証情報を検証します。認証情報が正しい場合のみユーザ情報を返却します。

TOPレベルエラー: USER\_NOT\_FOUND, USER\_EXPIRED, INVALID\_PASSWORD

戻り値: User!

引数:

引数	型	説明
input	<u>VerifyUserInput!</u>	

使用例:

```
query (
```

```
$input: VerifyUserInput!  
{  
  verifyUser(input: $input) {  
    # フィールドを指定  
  }  
}
```

## 2. ミューテーション (Mutation)

### 2-1. updateCloud

クラウドの設定情報を更新します。

戻り値: Cloud!

引数:

引数	型	説明
input	<u>UpdateCloudInput!</u>	

使用例:

```
mutation (
  $input: UpdateCloudInput!
){
  updateCloud(input: $input) {
    # フィールドを指定
  }
}
```

### 2-2. updateDateTime

現在時刻を設定、または、NTPサーバの設定をおこないます。

戻り値: DateTime!

引数:

引数	型	説明
input	<u>UpdateDateTimeInput!</u>	

使用例:

```
mutation (
  $input: UpdateDateTimeInput!
){
  updateDateTime(input: $input) {
    # フィールドを指定
  }
}
```

## 2-3. createEvent

イベント情報を作成します。

TOPレベルエラー: EVENT\_TRIGGER\_NOT\_FOUND, NOTIFICATION\_TARGET\_NOT\_FOUND

戻り値: Event!

引数:

引数	型	説明
input	<u>CreateEventInput!</u>	

使用例:

```
mutation (
  $input: CreateEventInput!
){
  createEvent(input: $input) {
    # フィールドを指定
  }
}
```

## 2-4. updateEvent

イベント情報を更新します。

TOPレベルエラー: EVENT\_NOT\_FOUND, EVENT\_TRIGGER\_NOT\_FOUND, NOTIFICATION\_TARGET\_NOT\_FOUND

戻り値: Event!

引数:

引数	型	説明
id	ID!	
input	<u>UpdateEventInput!</u>	

使用例:

```
mutation (
  $id: ID!,
  $input: UpdateEventInput!
){
  updateEvent(id: $id, input: $input) {
    # フィールドを指定
  }
}
```

## 2-5. deleteEvent

イベント情報を削除します。

TOPレベルエラー: EVENT\_NOT\_FOUND

戻り値: Event!

引数:

引数	型	説明
id	ID!	

使用例:

```
mutation (
  $id: ID!
){
  deleteEvent(id: $id) {
    # フィールドを指定
  }
}
```

## 2-6. testEvent

イベントをテスト実行します。

TOPレベルエラー: EVENT\_NOT\_FOUND

戻り値: Event!

引数:

引数	型	説明
id	ID!	

使用例:

```
mutation (
  $id: ID!
){
  testEvent(id: $id) {
    # フィールドを指定
  }
}
```

## 2-7. createEventTrigger

イベントトリガーを作成します。

戻り値: EventTrigger!

引数:

引数	型	説明
input	<u>CreateEventTriggerInput!</u>	

使用例:

```
mutation (
  $input: CreateEventTriggerInput!
){
  createEventTrigger(input: $input) {
    # フィールドを指定
  }
}
```

## 2-8. updateEventTrigger

イベントトリガーを更新します。

TOPレベルエラー: EVENT\_TRIGGER\_NOT\_FOUND

戻り値: EventTrigger!

引数:

引数	型	説明
id	ID!	
input	<u>UpdateEventTriggerInput!</u>	

使用例:

```
mutation (
  $id: ID!,
  $input: UpdateEventTriggerInput!
){
  updateEventTrigger(id: $id, input: $input) {
    # フィールドを指定
  }
}
```

## 2-9. deleteEventTrigger

イベントトリガーを削除します。

TOPレベルエラー: EVENT\_TRIGGER\_NOT\_FOUND, CURRENTLY\_IN\_USE

戻り値: EventTrigger!

引数:

引数	型	説明
id	ID!	

使用例:

```
mutation (
  $id: ID!
){
  deleteEventTrigger(id: $id) {
    # フィールドを指定
  }
}
```

## 2-10. updateInport

入力ポートの設定を更新します。

TOPレベルエラー: INPORT\_NOT\_FOUND

戻り値: Inport!

引数:

引数	型	説明
portNo	<u>InportPortNoEnum</u> !	
input	<u>UpdateInportInput</u> !	

使用例:

```
mutation (
  $portNo: InportPortNoEnum!,
  $input: UpdateInportInput!
){
  updateInport(portNo: $portNo, input: $input) {
    # フィールドを指定
  }
}
```

## 2-11. updateNetwork

ネットワークの設定を更新します。

戻り値: Network!

引数:

引数	型	説明
input	<u>UpdateNetworkInput</u>	

使用例:

```
mutation (
  $input: UpdateNetworkInput
){
  updateNetwork(input: $input) {
    # フィールドを指定
  }
}
```

## 2-12. createNotificationTarget

通知先設定を新規登録します。

戻り値: NotificationTarget!

引数:

引数	型	説明
input	<u>CreateNotificationTargetInput!</u>	

使用例:

```
mutation (
  $input: CreateNotificationTargetInput!
){
  createNotificationTarget(input: $input) {
    # フィールドを指定
  }
}
```

## 2-13. updateNotificationTarget

通知先設定を更新します。

TOPレベルエラー: NOTIFICATION\_TARGET\_NOT\_FOUND

戻り値: NotificationTarget!

引数:



引数	型	説明
id	ID!	
input	<u>UpdateNotificationTargetInput!</u>	

使用例:

```
mutation (
  $id: ID!,
  $input: UpdateNotificationTargetInput!
){
  updateNotificationTarget(id: $id, input: $input) {
    # フィールドを指定
  }
}
```

## 2-14. deleteNotificationTarget

通知先設定を削除します。

TOPレベルエラー: NOTIFICATION\_TARGET\_NOT\_FOUND, CURRENTLY\_IN\_USE

戻り値: NotificationTarget!

引数:

引数	型	説明
id	ID!	

使用例:

```
mutation (
  $id: ID!
){
  deleteNotificationTarget(id: $id) {
    # フィールドを指定
  }
}
```

## 2-15. notify

通知のテストを実行します。

TOPレベルエラー: NOTIFICATION\_TARGET\_NOT\_FOUND, NOTIFICATION\_FAILED

戻り値: NotificationTarget!

引数:

引数	型	説明
notificationTargetId	ID!	
input	<u>NotifyInput</u> !	

使用例:

```
mutation (
  $notificationTargetId: ID!,
  $input: NotifyInput!
){
  notify(notificationTargetId: $notificationTargetId, input: $input) {
    # フィールドを指定
  }
}
```

## 2-16. updateOutput

出力ポートの設定を更新します。

TOPレベルエラー: OUTPUT\_NOT\_FOUND

戻り値: Output!

引数:

引数	型	説明
portNo	<u>OutputPortNoEnum</u> !	
input	<u>UpdateOutputInput</u> !	

使用例:

```
mutation (
  $portNo: OutputPortNoEnum!,
  $input: UpdateOutputInput!
){
  updateOutput(portNo: $portNo, input: $input) {
    # フィールドを指定
  }
}
```

## 2-17. writeOutput

出力ポートの制御を実行します。

TOPレベルエラー: OUTPUT\_NOT\_FOUND

戻り値: Outport!

引数:

引数	型	説明
portNo	<u>OutportPortNoEnum!</u>	
input	<u>WriteOutportInput!</u>	

使用例:

```
mutation (
  $portNo: OutportPortNoEnum!,
  $input: WriteOutportInput!
){
  writeOutport(portNo: $portNo, input: $input) {
    # フィールドを指定
  }
}
```

## 2-18. onlineFirmwareUpdate

オンラインアップデートを実行します。

戻り値: OnlineFirmwareUpdateResponse!

引数:

引数	型	説明
input	<u>OnlineFirmwareUpdateInput</u>	

使用例:

```
mutation (
  $input: OnlineFirmwareUpdateInput
){
  onlineFirmwareUpdate(input: $input) {
    # フィールドを指定
  }
}
```

## 2-19. reboot

本機のOSを再起動します。

戻り値: RebootResponse!

使用例:

```
mutation{
  reboot{
    # フィールドを指定
  }
}
```

## 2-20. factoryReset

本機の設定とデータをリセットします。

戻り値: FactoryResetResponse!

使用例:

```
mutation{
  factoryReset{
    # フィールドを指定
  }
}
```

## 2-21. createUser

ユーザを新規登録します。

TOPレベルエラー: USER\_ID\_ALREADY\_EXISTS

戻り値: User!

引数:

引数	型	説明
input	<u>CreateUserInput!</u>	

使用例:

```
mutation (
  $input: CreateUserInput!
){
  createUser(input: $input) {
    # フィールドを指定
  }
}
```

## 2-22. updateUser

ユーザ情報を更新します。

TOPレベルエラー: USER\_NOT\_FOUND, USER\_ID\_ALREADY\_EXISTS

戻り値: User!

引数:

引数	型	説明
id	ID!	
input	<u>UpdateUserInput</u> !	

使用例:

```
mutation (
  $id: ID!,
  $input: UpdateUserInput!
){
  updateUser(id: $id, input: $input) {
    # フィールドを指定
  }
}
```

## 2-23. deleteUser

ユーザを削除します。

TOPレベルエラー: USER\_NOT\_FOUND

戻り値: User!

引数:

引数	型	説明
id	ID!	

使用例:

```
mutation (
  $id: ID!
){
  deleteUser(id: $id) {
    # フィールドを指定
  }
}
```

## 3. 型定義

### 3-1. オブジェクト型

#### 3-1-1. Cloud

クラウドの接続状態及び、設定情報

型: オブジェクト

フィールド:

フィールド	型	説明
kind	<u>CloudKindEnum!</u>	クラウドの種別
state	<u>CloudStateEnum!</u>	クラウドの接続状態
config	<u>CloudConfigUnion</u>	クラウドの設定情報

#### 3-1-2. CloudAzureConfig

Azure IoT Hubを使用する場合の設定情報

型: オブジェクト

フィールド:

フィールド	型	説明
connectionString	String!	Azure IoT Hub上で発行される接続文字列

#### 3-1-3. CloudSmartioConfig

SMARTIOクラウドを使用する場合の設定情報

型: オブジェクト

フィールド:

フィールド	型	説明
connectionString	String!	SMARTIOクラウド上で発行される接続文字列

#### 3-1-4. DateTime

型: オブジェクト

フィールド:

フィールド	型	説明
currentDateTime	<u>Time</u> !	現在時刻
config	<u>DateTimeConfig</u> !	時刻設定

### 3-1-5. DateTimeConfig

型: オブジェクト

フィールド:

フィールド	型	説明
ntp	String!	NTPサーバのホスト名またはIPアドレス

### 3-1-6. DeviceInfo

型: オブジェクト

フィールド:

フィールド	型	説明
modelName	String!	デバイスの型番
firmware	<u>DeviceFirmware</u> !	Firmwareのバージョン
upTime	<u>UpTime</u> !	起動してからの経過時間

### 3-1-7. DeviceFirmware

型: オブジェクト

フィールド:

フィールド	型	説明
channel	String!	オンラインアップデート時に参照するチャンネル
※内部で利用されます		
name	String!	Firmwareのバージョン番号「0.0.0」など

### 3-1-8. UpTime

型: オブジェクト

フィールド:

フィールド	型	説明
days	Int!	起動してからの経過時間(日)

フィールド	型	説明
hours	Int!	起動してからの経過時間(時)
minutes	Int!	起動してからの経過時間(分)
seconds	Int!	起動してからの経過時間(秒)

### 3-1-9. Event

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventTriggerID	ID!	
name	String!	
enabled	Boolean!	
eventTrigger	<u>EventTrigger</u> !	
notificationEvent	<u>EventNotification</u>	
writeRemoteOutportEvent	<u>EventWriteRemoteOutport</u>	

### 3-1-10. EventConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>EventEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-11. EventEdge

An edge in a connection.

型: オブジェクト

フィールド:



フィールド	型	説明
node	<u>Event</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-12. EventNotification

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventID	ID!	
event	<u>Event</u> !	
notificationTargets	[ <u>NotificationTarget</u> ]!	

### 3-1-13. EventNotificationConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>EventNotificationEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-14. EventNotificationEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>EventNotification</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-15. EventTrigger

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
name	String!	
eventTriggerKind	<u>EventTriggerKindEnum</u> !	
events	[ <u>Event</u> !]	
trigger	<u>EventTriggerUnion</u> !	

### 3-1-16. EventTriggerConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>EventTriggerEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-17. EventTriggerEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>EventTrigger</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-18. EventTriggerInport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventTriggerID	ID!	

フィールド	型	説明
condition	<u>InportEventTriggerConditionEnum!</u>	
inportState	<u>InportEventTriggerInportStateEnum!</u>	
eventTrigger	<u>EventTrigger!</u>	
inports	[ <u>Inport!</u> ]	

### 3-1-19. EventTriggerOutport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventTriggerID	ID!	
condition	<u>OutportEventTriggerConditionEnum!</u>	
outportState	<u>OutportEventTriggerOutportStateEnum!</u>	
eventTrigger	<u>EventTrigger!</u>	
outports	[ <u>Outport!</u> ]	

### 3-1-20. EventTriggerSchedule

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventTriggerID	ID!	
dateKind	<u>ScheduleEventTriggerDateKindEnum!</u>	
eventTrigger	<u>EventTrigger!</u>	
date	<u>ScheduleEventTriggerDate!</u>	
time	<u>ScheduleEventTriggerTime!</u>	

### 3-1-21. EventWriteRemoteOutport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventID	ID!	
event	<u>Event</u> !	
remoteOutputs	[ <u>RemoteOutput</u> !]	

### 3-1-22. EventWriteRemoteOutputConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>EventWriteRemoteOutputEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-23. EventWriteRemoteOutputEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>EventWriteRemoteOutput</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-24. Inport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
portNo	<u>InportPortNoEnum</u> !	
name	String!	
chatterThresholdMillisec	Int!	

フィールド	型	説明
inportEventTriggers	[ <a href="#">EventTriggerInport</a> !]	
state	<a href="#">InportStateEnum</a> !	

### 3-1-25. InportConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <a href="#">InportEdge</a> ]	A list of edges.
pageInfo	<a href="#">PageInfo</a> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-26. InportEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<a href="#">Inport</a>	The item at the end of the edge.
cursor	<a href="#">Cursor</a> !	A cursor for use in pagination.

### 3-1-27. LogEvent

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
eventID	ID	
eventTriggerID	ID	
eventTriggerKind	String!	
eventTriggerName	String!	
eventName	String!	

フィールド	型	説明
eventDetail	<u>Map</u>	
createdAt	<u>Time</u> !	

### 3-1-28. LogEventConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>LogEventEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-29. LogEventEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>LogEvent</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-30. LogInport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
portNo	<u>InportPortNoEnum</u> !	
portName	String!	
inputValue	String!	
createdAt	<u>Time</u> !	

### 3-1-31. LogInportConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <a href="#">LogInportEdge</a> ]	A list of edges.
pageInfo	<a href="#">PageInfo</a> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-32. LogInportEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<a href="#">LogInport</a>	The item at the end of the edge.
cursor	<a href="#">Cursor</a> !	A cursor for use in pagination.

### 3-1-33. LogOutport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
portNo	<a href="#">OutportPortNoEnum</a> !	
portName	String!	
operation	String!	
oneshotDurationMillisec	Int	
message	String!	
createdAt	<a href="#">Time</a> !	

### 3-1-34. LogOutportConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <a href="#">LogOutletEdge</a> ]	A list of edges.
pageInfo	<a href="#">PageInfo</a> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-35. LogOutletEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<a href="#">LogOutlet</a>	The item at the end of the edge.
cursor	<a href="#">Cursor</a> !	A cursor for use in pagination.

### 3-1-36. NotificationTarget

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
name	String!	
notificationTargetKind	<a href="#">NotificationTargetKindEnum</a> !	
notificationEvents	[ <a href="#">EventNotification</a> ! ]	
target	<a href="#">NotificationTargetUnion</a> !	

### 3-1-37. NotificationTargetConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <a href="#">NotificationTargetEdge</a> ]	A list of edges.
pageInfo	<a href="#">PageInfo</a> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.



### 3-1-38. NotificationTargetEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<a href="#">NotificationTarget</a>	The item at the end of the edge.
cursor	<a href="#">Cursor</a> !	A cursor for use in pagination.

### 3-1-39. NotificationTargetEmail

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
toAddressList	[String!]!	
fromAddress	String!	
fromName	String!	
smtpHost	String!	
smtpPort	Int!	
smtpAuth	<a href="#">SMTPAuthEnum</a> !	
smtpUser	String	
smtpPassword	String	
smtpEncryption	<a href="#">SMTPEncryptionEnum</a> !	
notificationtarget	<a href="#">NotificationTarget</a> !	

### 3-1-40. NotificationTargetWebhook

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
url	String!	

フィールド	型	説明
signingPassphrase	String	
notificationtarget	<u>NotificationTarget</u> !	

### 3-1-41. Outport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
portNo	<u>OutportPortNoEnum</u> !	
name	String!	
initialStateOnStartup	<u>OutportInitialStateOnStartupEnum</u> !	
outportEventTriggers	[ <u>EventTriggerOutport</u> !]	
state	<u>OutportStateEnum</u> !	

### 3-1-42. OutportConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>OutportEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-43. OutportEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>Outport</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-44. PageInfo

Information about pagination in a connection. <https://relay.dev/graphql/connections.htm#sec-undefined.PageInfo>

型: オブジェクト

フィールド:

フィールド	型	説明
hasNextPage	Boolean!	When paginating forwards, are there more items?
hasPreviousPage	Boolean!	When paginating backwards, are there more items?
startCursor	<u>Cursor</u>	When paginating backwards, the cursor to continue.
endCursor	<u>Cursor</u>	When paginating forwards, the cursor to continue.

### 3-1-45. RemoteOutputport

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
host	String!	
portNo	<u>OutputportPortNoEnum</u> !	
authUser	String!	
authPassword	String!	
operation	<u>RemoteOutputportOperationEnum</u> !	
oneshotDurationMillisec	Int	
writeRemoteOutputportEvent	[ <u>EventWriteRemoteOutputport</u> !]	

### 3-1-46. User

型: オブジェクト

フィールド:

フィールド	型	説明
id	ID!	
userID	String!	
userKind	<u>UserKindEnum</u> !	

フィールド	型	説明
memo	String!	
expiredAt	<u>Time</u>	
createdAt	<u>Time</u> !	
updatedAt	<u>Time</u> !	

### 3-1-47. UserConnection

A connection to a list of items.

型: オブジェクト

フィールド:

フィールド	型	説明
edges	[ <u>UserEdge</u> ]	A list of edges.
pageInfo	<u>PageInfo</u> !	Information to aid in pagination.
totalCount	Int!	Identifies the total count of items in the connection.

### 3-1-48. UserEdge

An edge in a connection.

型: オブジェクト

フィールド:

フィールド	型	説明
node	<u>User</u>	The item at the end of the edge.
cursor	<u>Cursor</u> !	A cursor for use in pagination.

### 3-1-49. EventTimelinesResponse

型: オブジェクト

フィールド:

フィールド	型	説明
timelines	[ <u>EventTimeline</u> ! ]!	

### 3-1-50. EventTimeline

型: オブジェクト

フィールド:

フィールド	型	説明
event	<u>Event</u> !	
eventTrigger	<u>EventTrigger</u> !	
time	<u>Time</u> !	

### 3-1-51. ScheduleEventTriggerDate

型: オブジェクト

フィールド:

フィールド	型	説明
kind	<u>ScheduleEventTriggerDateKindEnum</u> !	
months	<u>ScheduleEventTriggerDateMonths</u> !	
dayOfWeeks	<u>ScheduleEventTriggerDateDayOfWeeks</u> !	

### 3-1-52. ScheduleEventTriggerDateMonths

型: オブジェクト

フィールド:

フィールド	型	説明
january	Boolean!	
february	Boolean!	
march	Boolean!	
april	Boolean!	
may	Boolean!	
june	Boolean!	
july	Boolean!	
august	Boolean!	
september	Boolean!	
october	Boolean!	
november	Boolean!	
december	Boolean!	

### 3-1-53. ScheduleEventTriggerDateDayOfWeeks

型: オブジェクト

フィールド:

フィールド	型	説明
monday	Boolean!	
tuesday	Boolean!	
wednesday	Boolean!	
thursday	Boolean!	
friday	Boolean!	
saturday	Boolean!	
sunday	Boolean!	

**3-1-54. ScheduleEventTriggerTime**

型: オブジェクト

フィールド:

フィールド	型	説明
start	String!	
repeat	<u>ScheduleEventTriggerTimeRepeat</u>	

**3-1-55. ScheduleEventTriggerTimeRepeat**

型: オブジェクト

フィールド:

フィールド	型	説明
until	String!	
intervalMinutes	Int!	

**3-1-56. Network**

型: オブジェクト

フィールド:

フィールド	型	説明
hostName	String!	現在のホスト名
defaultGateway	String	現在のデフォルトゲートウェイ

フィールド	型	説明
dns	[String]!	現在のDNSサーバーリスト
ipv4address	String	現在のIPアドレス
macAddress	String!	macアドレス
config	<u>NetworkConfigUnion</u> !	設定情報

### 3-1-57. NetworkConfigDhcp

DHCP設定時のネットワーク設定

型: オブジェクト

フィールド:

フィールド	型	説明
primaryDns	String	プライマリDNSサーバー
secondaryDns	String	セカンダリDNSサーバー

### 3-1-58. NetworkConfigStatic

静的アドレス設定時のネットワーク設定

型: オブジェクト

フィールド:

フィールド	型	説明
ipv4Address	String!	IPアドレス
subnetMask	Int!	サブネットマスク
gateway	String!	ゲートウェイ
primaryDns	String!	プライマリDNSサーバー
secondaryDns	String	セカンダリDNSサーバー

### 3-1-59. LatestFirmware

最新ファームウェア情報

型: オブジェクト

フィールド:

フィールド	型	説明
channel	String!	ファームウェアのチャンネル

フィールド	型	説明
name	String!	最新ファームウェアバージョン名
url	String!	最新ファームウェアのダウンロードURL

### 3-1-60. OnlineFirmwareUpdateResponse

型: オブジェクト

フィールド:

フィールド	型	説明
_dummy	<u>Dummy</u>	

### 3-1-61. RebootResponse

型: オブジェクト

フィールド:

フィールド	型	説明
_dummy	<u>Dummy</u>	

### 3-1-62. FactoryResetResponse

型: オブジェクト

フィールド:

フィールド	型	説明
_dummy	<u>Dummy</u>	

## 3-2. インターフェース型

### 3-2-1. Node

An object with an ID. Follows the [Relay Global Object Identification Specification](#)

型: インターフェース

## 3-3. 入力型

### 3-3-1. UpdateCloudInput

クラウド設定更新時の入力データ

型: 入力オブジェクト



入力フィールド:

入力フィールド	型	説明
oneofCloudConfig	<u>OneofCloudConfigInput</u>	

### 3-3-2. OneofCloudConfigInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
azure	<u>CloudConfigAzureInput</u>	Azure IoT Hubを使用する場合の設定情報
smartio	<u>CloudConfigSmartioInput</u>	SMARTIOクラウドを使用する場合の設定情報
none	<u>CloudConfigNoneInput</u>	クラウドを使用しない場合

### 3-3-3. CloudConfigAzureInput

Azure IoT Hubを使用する場合の設定の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
connectionString	String!	Azure IoT Hub上で発行される接続文字列

### 3-3-4. CloudConfigSmartioInput

SMARTIOクラウドを使用する場合の設定の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
connectionString	String!	SMARTIOクラウド上で発行される接続文字列

### 3-3-5. CloudConfigNoneInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
_dummy	<u>Dummy</u>	

### 3-3-6. UpdateDateTimeInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
currentDateTime	<u>Time</u>	現在時刻を設定
config	<u>DateTimeConfigInput</u>	日時設定

### 3-3-7. DateTimeConfigInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
ntp	String!	NTPサーバのホスト名またはIPアドレス

### 3-3-8. EventNotificationOrder

Ordering options for EventNotification connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection!</u>	The ordering direction.
field	<u>EventNotificationOrderField!</u>	The field by which to order EventNotifications.

### 3-3-9. EventNotificationWhereInput

EventNotificationWhereInput is used for filtering EventNotification objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>EventNotificationWhereInput</u>	
and	[ <u>EventNotificationWhereInput!</u> ]	
or	[ <u>EventNotificationWhereInput!</u> ]	
id	ID	id field predicates
idNEQ	ID	

入力フィールド	型	説明
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventID	ID	event_id field predicates
eventIDNEQ	ID	
eventIDIn	[ID!]	
eventIDNotIn	[ID!]	
eventIDGT	ID	
eventIDGTE	ID	
eventIDLT	ID	
eventIDLTE	ID	
eventIDContains	ID	
eventIDHasPrefix	ID	
eventIDHasSuffix	ID	
eventIDEqualFold	ID	
eventIDContainsFold	ID	
hasEvent	Boolean	event edge predicates
hasEventWith	[ <u>EventWhereInput</u> !]	
hasNotificationTargets	Boolean	notificationTargets edge predicates
hasNotificationTargetsWith	[ <u>NotificationTargetWhereInput</u> !]	

### 3-3-10. EventOrder

Ordering options for Event connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>EventOrderField</u> !	The field by which to order Events.

### 3-3-11. EventTriggerInportOrder

Ordering options for EventTriggerInport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>EventTriggerInportOrderField</u> !	The field by which to order EventTriggerInports.

### 3-3-12. EventTriggerInportWhereInput

EventTriggerInportWhereInput is used for filtering EventTriggerInport objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>EventTriggerInportWhereInput</u>	
and	[ <u>EventTriggerInportWhereInput</u> !]	
or	[ <u>EventTriggerInportWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventTriggerID	ID	event_trigger_id field predicates
eventTriggerIDNEQ	ID	

入力フィールド	型	説明
eventTriggerIDIn	[ID!]	
eventTriggerIDNotIn	[ID!]	
eventTriggerIDGT	ID	
eventTriggerIDGTE	ID	
eventTriggerIDLT	ID	
eventTriggerIDLTE	ID	
eventTriggerIDContains	ID	
eventTriggerIDHasPrefix	ID	
eventTriggerIDHasSuffix	ID	
eventTriggerIDEqualFold	ID	
eventTriggerIDContainsFold	ID	
condition	<u>InportEventTriggerConditionEnum</u>	condition field predicates
conditionNEQ	<u>InportEventTriggerConditionEnum</u>	
conditionIn	[ <u>InportEventTriggerConditionEnum</u> !]	
conditionNotIn	[ <u>InportEventTriggerConditionEnum</u> !]	
inportState	<u>InportEventTriggerInportStateEnum</u>	inport_state field predicates
inportStateNEQ	<u>InportEventTriggerInportStateEnum</u>	
inportStateIn	[ <u>InportEventTriggerInportStateEnum</u> !]	
inportStateNotIn	[ <u>InportEventTriggerInportStateEnum</u> !]	
hasEventTrigger	Boolean	eventTrigger edge predicates
hasEventTriggerWith	[ <u>EventTriggerWhereInput</u> !]	
hasInports	Boolean	inports edge predicates
hasInportsWith	[ <u>InportWhereInput</u> !]	

### 3-3-13. EventTriggerOrder

Ordering options for EventTrigger connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>EventTriggerOrderField</u> !	The field by which to order EventTriggers.

### 3-3-14. EventTriggerOutportOrder

Ordering options for EventTriggerOutport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>EventTriggerOutportOrderField</u> !	The field by which to order EventTriggerOutports.

### 3-3-15. EventTriggerOutportWhereInput

EventTriggerOutportWhereInput is used for filtering EventTriggerOutport objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>EventTriggerOutportWhereInput</u>	
and	[ <u>EventTriggerOutportWhereInput</u> !]	
or	[ <u>EventTriggerOutportWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventTriggerID	ID	event_trigger_id field predicates
eventTriggerIDNEQ	ID	

入力フィールド	型	説明
eventTriggerIDIn	[ID!]	
eventTriggerIDNotIn	[ID!]	
eventTriggerIDGT	ID	
eventTriggerIDGTE	ID	
eventTriggerIDLT	ID	
eventTriggerIDLTE	ID	
eventTriggerIDContains	ID	
eventTriggerIDHasPrefix	ID	
eventTriggerIDHasSuffix	ID	
eventTriggerIDEqualFold	ID	
eventTriggerIDContainsFold	ID	
condition	<u>OutputEventTriggerConditionEnum</u>	condition field predicates
conditionNEQ	<u>OutputEventTriggerConditionEnum</u>	
conditionIn	[ <u>OutputEventTriggerConditionEnum</u> !]	
conditionNotIn	[ <u>OutputEventTriggerConditionEnum</u> !]	
outputState	<u>OutputEventTriggerOutputStateEnum</u>	output_state field predicates
outputStateNEQ	<u>OutputEventTriggerOutputStateEnum</u>	
outputStateIn	[ <u>OutputEventTriggerOutputStateEnum</u> !]	
outputStateNotIn	[ <u>OutputEventTriggerOutputStateEnum</u> !]	
hasEventTrigger	Boolean	eventTrigger edge predicates
hasEventTriggerWith	[ <u>EventTriggerWhereInput</u> !]	
hasOutputs	Boolean	outputs edge predicates
hasOutputsWith	[ <u>OutputWhereInput</u> !]	

### 3-3-16. EventTriggerScheduleOrder

Ordering options for EventTriggerSchedule connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection!</u>	The ordering direction.
field	<u>EventTriggerScheduleOrderField!</u>	The field by which to order EventTriggerSchedules.

### 3-3-17. EventTriggerScheduleWhereInput

EventTriggerScheduleWhereInput is used for filtering EventTriggerSchedule objects. Input was generated by ent.

**型:** 入力オブジェクト

**入力フィールド:**

入力フィールド	型	説明
not	<u>EventTriggerScheduleWhereInput</u>	
and	[ <u>EventTriggerScheduleWhereInput!</u> ]	
or	[ <u>EventTriggerScheduleWhereInput!</u> ]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventTriggerID	ID	event_trigger_id field predicates
eventTriggerIDNEQ	ID	
eventTriggerIDIn	[ID!]	
eventTriggerIDNotIn	[ID!]	
eventTriggerIDGT	ID	
eventTriggerIDGTE	ID	
eventTriggerIDLT	ID	
eventTriggerIDLTE	ID	
eventTriggerIDContains	ID	
eventTriggerIDHasPrefix	ID	



入力フィールド	型	説明
eventTriggerIDHasSuffix	ID	
eventTriggerIDEqualFold	ID	
eventTriggerIDContainsFold	ID	
dateKind	<a href="#">ScheduleEventTriggerDateKindEnum</a>	date_kind field predicates
dateKindNEQ	<a href="#">ScheduleEventTriggerDateKindEnum</a>	
dateKindIn	[ <a href="#">ScheduleEventTriggerDateKindEnum</a> !]	
dateKindNotIn	[ <a href="#">ScheduleEventTriggerDateKindEnum</a> !]	
hasEventTrigger	Boolean	eventTrigger edge predicates
hasEventTriggerWith	[ <a href="#">EventTriggerWhereInput</a> !]	

### 3-3-18. EventTriggerWhereInput

EventTriggerWhereInput is used for filtering EventTrigger objects. Input was generated by ent.

**型:** 入力オブジェクト

**入力フィールド:**

入力フィールド	型	説明
not	<a href="#">EventTriggerWhereInput</a>	
and	[ <a href="#">EventTriggerWhereInput</a> !]	
or	[ <a href="#">EventTriggerWhereInput</a> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
name	String	name field predicates
nameNEQ	String	
nameIn	[String!]	

入力フィールド	型	説明
nameNotIn	[String!]	
nameGT	String	
nameGTE	String	
nameLT	String	
nameLTE	String	
nameContains	String	
nameHasPrefix	String	
nameHasSuffix	String	
nameEqualFold	String	
nameContainsFold	String	
eventTriggerKind	<u>EventTriggerKindEnum</u>	event_trigger_kind field predicates
eventTriggerKindNEQ	<u>EventTriggerKindEnum</u>	
eventTriggerKindIn	[ <u>EventTriggerKindEnum</u> !]	
eventTriggerKindNotIn	[ <u>EventTriggerKindEnum</u> !]	
hasEvents	Boolean	events edge predicates
hasEventsWith	[ <u>EventWhereInput</u> !]	

### 3-3-19. EventWhereInput

EventWhereInput is used for filtering Event objects. Input was generated by ent.

**型:** 入力オブジェクト

**入力フィールド:**

入力フィールド	型	説明
not	<u>EventWhereInput</u>	
and	[ <u>EventWhereInput</u> !]	
or	[ <u>EventWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	

入力フィールド	型	説明
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventTriggerID	ID	event_trigger_id field predicates
eventTriggerIDNEQ	ID	
eventTriggerIDIn	[ID!]	
eventTriggerIDNotIn	[ID!]	
eventTriggerIDGT	ID	
eventTriggerIDGTE	ID	
eventTriggerIDLT	ID	
eventTriggerIDLTE	ID	
eventTriggerIDContains	ID	
eventTriggerIDHasPrefix	ID	
eventTriggerIDHasSuffix	ID	
eventTriggerIDEqualFold	ID	
eventTriggerIDContainsFold	ID	
name	String	name field predicates
nameNEQ	String	
nameIn	[String!]	
nameNotIn	[String!]	
nameGT	String	
nameGTE	String	
nameLT	String	
nameLTE	String	
nameContains	String	
nameHasPrefix	String	
nameHasSuffix	String	

入力フィールド	型	説明
nameEqualFold	String	
nameContainsFold	String	
enabled	Boolean	enabled field predicates
enabledNEQ	Boolean	
hasEventTrigger	Boolean	eventTrigger edge predicates
hasEventTriggerWith	[ <a href="#">EventTriggerWhereInput!</a> ]	
hasNotificationEvent	Boolean	notificationEvent edge predicates
hasNotificationEventWith	[ <a href="#">EventNotificationWhereInput!</a> ]	
hasWriteRemoteOutputEvent	Boolean	writeRemoteOutputEvent edge predicates
hasWriteRemoteOutputEventWith	[ <a href="#">EventWriteRemoteOutputWhereInput!</a> ]	

### 3-3-20. EventWriteRemoteOutputOrder

Ordering options for EventWriteRemoteOutput connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection!</a>	The ordering direction.
field	<a href="#">EventWriteRemoteOutputOrderField!</a>	The field by which to order EventWriteRemoteOutputs.

### 3-3-21. EventWriteRemoteOutputWhereInput

EventWriteRemoteOutputWhereInput is used for filtering EventWriteRemoteOutput objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<a href="#">EventWriteRemoteOutputWhereInput</a>	
and	[ <a href="#">EventWriteRemoteOutputWhereInput!</a> ]	
or	[ <a href="#">EventWriteRemoteOutputWhereInput!</a> ]	
id	ID	id field predicates
idNEQ	ID	

入力フィールド	型	説明
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventID	ID	event_id field predicates
eventIDNEQ	ID	
eventIDIn	[ID!]	
eventIDNotIn	[ID!]	
eventIDGT	ID	
eventIDGTE	ID	
eventIDLT	ID	
eventIDLTE	ID	
eventIDContains	ID	
eventIDHasPrefix	ID	
eventIDHasSuffix	ID	
eventIDEqualFold	ID	
eventIDContainsFold	ID	
hasEvent	Boolean	event edge predicates
hasEventWith	[ <a href="#">EventWhereInput!</a> ]	
hasRemoteOutputs	Boolean	remoteOutputs edge predicates
hasRemoteOutputsWith	[ <a href="#">RemoteOutputWhereInput!</a> ]	

### 3-3-22. InportOrder

Ordering options for Inport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>InportOrderField</u> !	The field by which to order Inports.

### 3-3-23. InportWhereInput

InportWhereInput is used for filtering Inport objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>InportWhereInput</u>	
and	[ <u>InportWhereInput</u> !]	
or	[ <u>InportWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
portNo	<u>InportPortNoEnum</u>	port_no field predicates
portNoNEQ	<u>InportPortNoEnum</u>	
portNoIn	[ <u>InportPortNoEnum</u> !]	
portNoNotIn	[ <u>InportPortNoEnum</u> !]	
name	String	name field predicates
nameNEQ	String	
nameIn	[String!]	
nameNotIn	[String!]	
nameGT	String	
nameGTE	String	

入力フィールド	型	説明
nameLT	String	
nameLTE	String	
nameContains	String	
nameHasPrefix	String	
nameHasSuffix	String	
nameEqualFold	String	
nameContainsFold	String	
chatterThresholdMillisec	Int	chatter_threshold_millisec field predicates
chatterThresholdMillisecNEQ	Int	
chatterThresholdMillisecIn	[Int!]	
chatterThresholdMillisecNotIn	[Int!]	
chatterThresholdMillisecGT	Int	
chatterThresholdMillisecGTE	Int	
chatterThresholdMillisecLT	Int	
chatterThresholdMillisecLTE	Int	
hasInportEventTriggers	Boolean	inportEventTriggers edge predicates
hasInportEventTriggersWith	[ <u>EventTriggerInportWhereInput</u> !]	

### 3-3-24. LogEventOrder

Ordering options for LogEvent connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>LogEventOrderField</u> !	The field by which to order LogEvents.

### 3-3-25. LogEventWhereInput

LogEventWhereInput is used for filtering LogEvent objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>LogEventWhereInput</u>	
and	[ <u>LogEventWhereInput</u> !]	
or	[ <u>LogEventWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
eventID	ID	event_id field predicates
eventIDNEQ	ID	
eventIDIn	[ID!]	
eventIDNotIn	[ID!]	
eventIDGT	ID	
eventIDGTE	ID	
eventIDLT	ID	
eventIDLTE	ID	
eventIDContains	ID	
eventIDHasPrefix	ID	
eventIDHasSuffix	ID	
eventIDIsNil	Boolean	
eventIDNotNil	Boolean	
eventIDEqualFold	ID	
eventIDContainsFold	ID	
eventTriggerID	ID	event_trigger_id field predicates
eventTriggerIDNEQ	ID	



入力フィールド	型	説明
eventTriggerIDIn	[ID!]	
eventTriggerIDNotIn	[ID!]	
eventTriggerIDGT	ID	
eventTriggerIDGTE	ID	
eventTriggerIDLT	ID	
eventTriggerIDLTE	ID	
eventTriggerIDContains	ID	
eventTriggerIDHasPrefix	ID	
eventTriggerIDHasSuffix	ID	
eventTriggerIDIsNil	Boolean	
eventTriggerIDNotNil	Boolean	
eventTriggerIDEqualFold	ID	
eventTriggerIDContainsFold	ID	
eventTriggerKind	String	event_trigger_kind field predicates
eventTriggerKindNEQ	String	
eventTriggerKindIn	[String!]	
eventTriggerKindNotIn	[String!]	
eventTriggerKindGT	String	
eventTriggerKindGTE	String	
eventTriggerKindLT	String	
eventTriggerKindLTE	String	
eventTriggerKindContains	String	
eventTriggerKindHasPrefix	String	
eventTriggerKindHasSuffix	String	
eventTriggerKindEqualFold	String	
eventTriggerKindContainsFold	String	
eventTriggerName	String	event_trigger_name field predicates
eventTriggerNameNEQ	String	

入力フィールド	型	説明
eventTriggerNameIn	[String!]	
eventTriggerNameNotIn	[String!]	
eventTriggerNameGT	String	
eventTriggerNameGTE	String	
eventTriggerNameLT	String	
eventTriggerNameLTE	String	
eventTriggerNameContains	String	
eventTriggerNameHasPrefix	String	
eventTriggerNameHasSuffix	String	
eventTriggerNameEqualFold	String	
eventTriggerNameContainsFold	String	
eventName	String	event_name field predicates
eventNameNEQ	String	
eventNameIn	[String!]	
eventNameNotIn	[String!]	
eventNameGT	String	
eventNameGTE	String	
eventNameLT	String	
eventNameLTE	String	
eventNameContains	String	
eventNameHasPrefix	String	
eventNameHasSuffix	String	
eventNameEqualFold	String	
eventNameContainsFold	String	
createdAt	<u>Time</u>	created_at field predicates
createdAtNEQ	<u>Time</u>	
createdAtIn	[ <u>Time</u> !]	
createdAtNotIn	[ <u>Time</u> !]	

入力フィールド	型	説明
createdAtGT	<u>Time</u>	
createdAtGTE	<u>Time</u>	
createdAtLT	<u>Time</u>	
createdAtLTE	<u>Time</u>	

### 3-3-26. LogInportOrder

Ordering options for LogInport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>LogInportOrderField</u> !	The field by which to order LogInports.

### 3-3-27. LogInportWhereInput

LogInportWhereInput is used for filtering LogInport objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>LogInportWhereInput</u>	
and	[ <u>LogInportWhereInput</u> !]	
or	[ <u>LogInportWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	

入力フィールド	型	説明
portNo	<u>InportPortNoEnum</u>	port_no field predicates
portNoNEQ	<u>InportPortNoEnum</u>	
portNoIn	[ <u>InportPortNoEnum</u> !]	
portNoNotIn	[ <u>InportPortNoEnum</u> !]	
portName	String	port_name field predicates
portNameNEQ	String	
portNameIn	[String!]	
portNameNotIn	[String!]	
portNameGT	String	
portNameGTE	String	
portNameLT	String	
portNameLTE	String	
portNameContains	String	
portNameHasPrefix	String	
portNameHasSuffix	String	
portNameEqualFold	String	
portNameContainsFold	String	
inputValue	String	input_value field predicates
inputValueNEQ	String	
inputValueIn	[String!]	
inputValueNotIn	[String!]	
inputValueGT	String	
inputValueGTE	String	
inputValueLT	String	
inputValueLTE	String	
inputValueContains	String	
inputValueHasPrefix	String	
inputValueHasSuffix	String	

入力フィールド	型	説明
inputValueEqualFold	String	
inputValueContainsFold	String	
createdAt	<u>Time</u>	created_at field predicates
createdAtNEQ	<u>Time</u>	
createdAtIn	[ <u>Time</u> !]	
createdAtNotIn	[ <u>Time</u> !]	
createdAtGT	<u>Time</u>	
createdAtGTE	<u>Time</u>	
createdAtLT	<u>Time</u>	
createdAtLTE	<u>Time</u>	

### 3-3-28. LogOutputOrder

Ordering options for LogOutput connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>LogOutputOrderField</u> !	The field by which to order LogOutputs.

### 3-3-29. LogOutputWhereInput

LogOutputWhereInput is used for filtering LogOutput objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>LogOutputWhereInput</u>	
and	[ <u>LogOutputWhereInput</u> !]	
or	[ <u>LogOutputWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	

入力フィールド	型	説明
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
portNo	<u>OutportPortNoEnum</u>	port_no field predicates
portNoNEQ	<u>OutportPortNoEnum</u>	
portNoIn	[ <u>OutportPortNoEnum</u> !]	
portNoNotIn	[ <u>OutportPortNoEnum</u> !]	
portName	String	port_name field predicates
portNameNEQ	String	
portNameIn	[String!]	
portNameNotIn	[String!]	
portNameGT	String	
portNameGTE	String	
portNameLT	String	
portNameLTE	String	
portNameContains	String	
portNameHasPrefix	String	
portNameHasSuffix	String	
portNameEqualFold	String	
portNameContainsFold	String	
operation	String	operation field predicates
operationNEQ	String	
operationIn	[String!]	
operationNotIn	[String!]	
operationGT	String	

入力フィールド	型	説明
operationGTE	String	
operationLT	String	
operationLTE	String	
operationContains	String	
operationHasPrefix	String	
operationHasSuffix	String	
operationEqualFold	String	
operationContainsFold	String	
oneshotDurationMillisec	Int	oneshot_duration_millisec field predicates
oneshotDurationMillisecNEQ	Int	
oneshotDurationMillisecIn	[Int!]	
oneshotDurationMillisecNotIn	[Int!]	
oneshotDurationMillisecGT	Int	
oneshotDurationMillisecGTE	Int	
oneshotDurationMillisecLT	Int	
oneshotDurationMillisecLTE	Int	
oneshotDurationMillisecIsNil	Boolean	
oneshotDurationMillisecNotNil	Boolean	
message	String	message field predicates
messageNEQ	String	
messageIn	[String!]	
messageNotIn	[String!]	
messageGT	String	
messageGTE	String	
messageLT	String	
messageLTE	String	
messageContains	String	
messageHasPrefix	String	

入力フィールド	型	説明
messageHasSuffix	String	
messageEqualFold	String	
messageContainsFold	String	
createdAt	<u>Time</u>	created_at field predicates
createdAtNEQ	<u>Time</u>	
createdAtIn	[ <u>Time</u> ! ]	
createdAtNotIn	[ <u>Time</u> ! ]	
createdAtGT	<u>Time</u>	
createdAtGTE	<u>Time</u>	
createdAtLT	<u>Time</u>	
createdAtLTE	<u>Time</u>	

### 3-3-30. NotificationTargetEmailOrder

Ordering options for NotificationTargetEmail connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<u>OrderDirection</u> !	The ordering direction.
field	<u>NotificationTargetEmailOrderField</u> !	The field by which to order NotificationTargetEmails.

### 3-3-31. NotificationTargetEmailWhereInput

NotificationTargetEmailWhereInput is used for filtering NotificationTargetEmail objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>NotificationTargetEmailWhereInput</u>	
and	[ <u>NotificationTargetEmailWhereInput</u> ! ]	
or	[ <u>NotificationTargetEmailWhereInput</u> ! ]	
id	ID	id field predicates



入力フィールド	型	説明
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
fromAddress	String	from_address field predicates
fromAddressNEQ	String	
fromAddressIn	[String!]	
fromAddressNotIn	[String!]	
fromAddressGT	String	
fromAddressGTE	String	
fromAddressLT	String	
fromAddressLTE	String	
fromAddressContains	String	
fromAddressHasPrefix	String	
fromAddressHasSuffix	String	
fromAddressEqualFold	String	
fromAddressContainsFold	String	
fromName	String	from_name field predicates
fromNameNEQ	String	
fromNameIn	[String!]	
fromNameNotIn	[String!]	
fromNameGT	String	
fromNameGTE	String	
fromNameLT	String	
fromNameLTE	String	

入力フィールド	型	説明
fromNameContains	String	
fromNameHasPrefix	String	
fromNameHasSuffix	String	
fromNameEqualFold	String	
fromNameContainsFold	String	
smtpHost	String	smtp_host field predicates
smtpHostNEQ	String	
smtpHostIn	[String!]	
smtpHostNotIn	[String!]	
smtpHostGT	String	
smtpHostGTE	String	
smtpHostLT	String	
smtpHostLTE	String	
smtpHostContains	String	
smtpHostHasPrefix	String	
smtpHostHasSuffix	String	
smtpHostEqualFold	String	
smtpHostContainsFold	String	
smtpPort	Int	smtp_port field predicates
smtpPortNEQ	Int	
smtpPortIn	[Int!]	
smtpPortNotIn	[Int!]	
smtpPortGT	Int	
smtpPortGTE	Int	
smtpPortLT	Int	
smtpPortLTE	Int	
smtpAuth	<u>SMTPAuthEnum</u>	smtp_auth field predicates
smtpAuthNEQ	<u>SMTPAuthEnum</u>	

入力フィールド	型	説明
smtpAuthIn	[ <a href="#">SMTPAuthEnum</a> !]	
smtpAuthNotIn	[ <a href="#">SMTPAuthEnum</a> !]	
smtpUser	String	smtp_user field predicates
smtpUserNEQ	String	
smtpUserIn	[String!]	
smtpUserNotIn	[String!]	
smtpUserGT	String	
smtpUserGTE	String	
smtpUserLT	String	
smtpUserLTE	String	
smtpUserContains	String	
smtpUserHasPrefix	String	
smtpUserHasSuffix	String	
smtpUserIsNil	Boolean	
smtpUserNotNil	Boolean	
smtpUserEqualFold	String	
smtpUserContainsFold	String	
smtpPassword	String	smtp_password field predicates
smtpPasswordNEQ	String	
smtpPasswordIn	[String!]	
smtpPasswordNotIn	[String!]	
smtpPasswordGT	String	
smtpPasswordGTE	String	
smtpPasswordLT	String	
smtpPasswordLTE	String	
smtpPasswordContains	String	
smtpPasswordHasPrefix	String	
smtpPasswordHasSuffix	String	

入力フィールド	型	説明
smtpPasswordIsNil	Boolean	
smtpPasswordNotNil	Boolean	
smtpPasswordEqualFold	String	
smtpPasswordContainsFold	String	
smtpEncryption	<a href="#">SMTPEncryptionEnum</a>	smtp_encryption field predicates
smtpEncryptionNEQ	<a href="#">SMTPEncryptionEnum</a>	
smtpEncryptionIn	[ <a href="#">SMTPEncryptionEnum</a> !]	
smtpEncryptionNotIn	[ <a href="#">SMTPEncryptionEnum</a> !]	
hasNotificationTarget	Boolean	notificationTarget edge predicates
hasNotificationTargetWith	[ <a href="#">NotificationTargetWhereInput</a> !]	

### 3-3-32. NotificationTargetOrder

Ordering options for NotificationTarget connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection</a> !	The ordering direction.
field	<a href="#">NotificationTargetOrderField</a> !	The field by which to order NotificationTargets.

### 3-3-33. NotificationTargetWebhookOrder

Ordering options for NotificationTargetWebhook connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection</a> !	The ordering direction.
field	<a href="#">NotificationTargetWebhookOrderField</a> !	The field by which to order NotificationTargetWebhooks.

### 3-3-34. NotificationTargetWebhookWhereInput

NotificationTargetWebhookWhereInput is used for filtering NotificationTargetWebhook objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>NotificationTargetWebhookWhereInput</u>	
and	[ <u>NotificationTargetWebhookWhereInput</u> !]	
or	[ <u>NotificationTargetWebhookWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
url	String	url field predicates
urlNEQ	String	
urlIn	[String!]	
urlNotIn	[String!]	
urlGT	String	
urlGTE	String	
urlLT	String	
urlLTE	String	
urlContains	String	
urlHasPrefix	String	
urlHasSuffix	String	
urlEqualFold	String	
urlContainsFold	String	
signingPassphrase	String	signing_passphrase field predicates
signingPassphraseNEQ	String	
signingPassphraseIn	[String!]	
signingPassphraseNotIn	[String!]	

入力フィールド	型	説明
signingPassphraseGT	String	
signingPassphraseGTE	String	
signingPassphraseLT	String	
signingPassphraseLTE	String	
signingPassphraseContains	String	
signingPassphraseHasPrefix	String	
signingPassphraseHasSuffix	String	
signingPassphraseIsNil	Boolean	
signingPassphraseNotNil	Boolean	
signingPassphraseEqualFold	String	
signingPassphraseContainsFold	String	
hasNotificationTarget	Boolean	notificationTarget edge predicates
hasNotificationTargetWith	[ <a href="#">NotificationTargetWhereInput</a> !]	

### 3-3-35. NotificationTargetWhereInput

NotificationTargetWhereInput is used for filtering NotificationTarget objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<a href="#">NotificationTargetWhereInput</a>	
and	[ <a href="#">NotificationTargetWhereInput</a> !]	
or	[ <a href="#">NotificationTargetWhereInput</a> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	

入力フィールド	型	説明
idLTE	ID	
name	String	name field predicates
nameNEQ	String	
nameIn	[String!]	
nameNotIn	[String!]	
nameGT	String	
nameGTE	String	
nameLT	String	
nameLTE	String	
nameContains	String	
nameHasPrefix	String	
nameHasSuffix	String	
nameEqualFold	String	
nameContainsFold	String	
notificationTargetKind	<a href="#">NotificationTargetKindEnum</a>	notification_target_kind field predicates
notificationTargetKindNEQ	<a href="#">NotificationTargetKindEnum</a>	
notificationTargetKindIn	[ <a href="#">NotificationTargetKindEnum</a> !]	
notificationTargetKindNotIn	[ <a href="#">NotificationTargetKindEnum</a> !]	
hasNotificationEvents	Boolean	notificationEvents edge predicates
hasNotificationEventsWith	[ <a href="#">EventNotificationWhereInput</a> !]	

### 3-3-36. OutportOrder

Ordering options for Outport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection</a> !	The ordering direction.
field	<a href="#">OutportOrderField</a> !	The field by which to order Outports.

### 3-3-37. OutportWhereInput

OutputWhereInput is used for filtering Output objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<u>OutputWhereInput</u>	
and	[ <u>OutputWhereInput</u> !]	
or	[ <u>OutputWhereInput</u> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
portNo	<u>OutputPortNoEnum</u>	port_no field predicates
portNoNEQ	<u>OutputPortNoEnum</u>	
portNoIn	[ <u>OutputPortNoEnum</u> !]	
portNoNotIn	[ <u>OutputPortNoEnum</u> !]	
name	String	name field predicates
nameNEQ	String	
nameIn	[String!]	
nameNotIn	[String!]	
nameGT	String	
nameGTE	String	
nameLT	String	
nameLTE	String	
nameContains	String	
nameHasPrefix	String	



入力フィールド	型	説明
nameHasSuffix	String	
nameEqualFold	String	
nameContainsFold	String	
initialStateOnStartup	<a href="#">OutportInitialStateOnStartupEnum</a>	initial_state_on_startup field predicates
initialStateOnStartupNEQ	<a href="#">OutportInitialStateOnStartupEnum</a>	
initialStateOnStartupIn	[ <a href="#">OutportInitialStateOnStartupEnum</a> !]	
initialStateOnStartupNotIn	[ <a href="#">OutportInitialStateOnStartupEnum</a> !]	
hasOutportEventTriggers	Boolean	outportEventTriggers edge predicates
hasOutportEventTriggersWith	[ <a href="#">EventTriggerOutportWhereInput</a> !]	

### 3-3-38. RemoteOutportOrder

Ordering options for RemoteOutport connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection</a> !	The ordering direction.
field	<a href="#">RemoteOutportOrderField</a> !	The field by which to order RemoteOutports.

### 3-3-39. RemoteOutportWhereInput

RemoteOutportWhereInput is used for filtering RemoteOutport objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<a href="#">RemoteOutportWhereInput</a>	
and	[ <a href="#">RemoteOutportWhereInput</a> !]	
or	[ <a href="#">RemoteOutportWhereInput</a> !]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	

入力フィールド	型	説明
idNotIn	[ID!]	
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
host	String	host field predicates
hostNEQ	String	
hostIn	[String!]	
hostNotIn	[String!]	
hostGT	String	
hostGTE	String	
hostLT	String	
hostLTE	String	
hostContains	String	
hostHasPrefix	String	
hostHasSuffix	String	
hostEqualFold	String	
hostContainsFold	String	
portNo	<u>OutportPortNoEnum</u>	port_no field predicates
portNoNEQ	<u>OutportPortNoEnum</u>	
portNoIn	[ <u>OutportPortNoEnum</u> !]	
portNoNotIn	[ <u>OutportPortNoEnum</u> !]	
authUser	String	auth_user field predicates
authUserNEQ	String	
authUserIn	[String!]	
authUserNotIn	[String!]	
authUserGT	String	
authUserGTE	String	

入力フィールド	型	説明
authUserLT	String	
authUserLTE	String	
authUserContains	String	
authUserHasPrefix	String	
authUserHasSuffix	String	
authUserEqualFold	String	
authUserContainsFold	String	
authPassword	String	auth_password field predicates
authPasswordNEQ	String	
authPasswordIn	[String!]	
authPasswordNotIn	[String!]	
authPasswordGT	String	
authPasswordGTE	String	
authPasswordLT	String	
authPasswordLTE	String	
authPasswordContains	String	
authPasswordHasPrefix	String	
authPasswordHasSuffix	String	
authPasswordEqualFold	String	
authPasswordContainsFold	String	
operation	<a href="#">RemoteOutportOperationEnum</a>	operation field predicates
operationNEQ	<a href="#">RemoteOutportOperationEnum</a>	
operationIn	[ <a href="#">RemoteOutportOperationEnum</a> !]	
operationNotIn	[ <a href="#">RemoteOutportOperationEnum</a> !]	
oneshotDurationMillisec	Int	oneshot_duration_millisec field predicates
oneshotDurationMillisecNEQ	Int	
oneshotDurationMillisecIn	[Int!]	
oneshotDurationMillisecNotIn	[Int!]	

入力フィールド	型	説明
oneshotDurationMillisecGT	Int	
oneshotDurationMillisecGTE	Int	
oneshotDurationMillisecLT	Int	
oneshotDurationMillisecLTE	Int	
oneshotDurationMillisecIsNil	Boolean	
oneshotDurationMillisecNotNil	Boolean	
hasWriteRemoteOutputEvent	Boolean	writeRemoteOutputEvent edge predicates
hasWriteRemoteOutputEventWith	[ <a href="#">EventWriteRemoteOutputWhereInput!</a> ]	

### 3-3-40. UserOrder

Ordering options for User connections

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
direction	<a href="#">OrderDirection!</a>	The ordering direction.
field	<a href="#">UserOrderField!</a>	The field by which to order Users.

### 3-3-41. UserWhereInput

UserWhereInput is used for filtering User objects. Input was generated by ent.

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
not	<a href="#">UserWhereInput</a>	
and	[ <a href="#">UserWhereInput!</a> ]	
or	[ <a href="#">UserWhereInput!</a> ]	
id	ID	id field predicates
idNEQ	ID	
idIn	[ID!]	
idNotIn	[ID!]	

入力フィールド	型	説明
idGT	ID	
idGTE	ID	
idLT	ID	
idLTE	ID	
userID	String	user_id field predicates
userIDNEQ	String	
userIDIn	[String!]	
userIDNotIn	[String!]	
userIDGT	String	
userIDGTE	String	
userIDLT	String	
userIDLTE	String	
userIDContains	String	
userIDHasPrefix	String	
userIDHasSuffix	String	
userIDEqualFold	String	
userIDContainsFold	String	
userKind	<u>UserKindEnum</u>	user_kind field predicates
userKindNEQ	<u>UserKindEnum</u>	
userKindIn	[ <u>UserKindEnum</u> !]	
userKindNotIn	[ <u>UserKindEnum</u> !]	
memo	String	memo field predicates
memoNEQ	String	
memoIn	[String!]	
memoNotIn	[String!]	
memoGT	String	
memoGTE	String	
memoLT	String	

入力フィールド	型	説明
memoLTE	String	
memoContains	String	
memoHasPrefix	String	
memoHasSuffix	String	
memoEqualFold	String	
memoContainsFold	String	
expiredAt	<u>Time</u>	expired_at field predicates
expiredAtNEQ	<u>Time</u>	
expiredAtIn	[ <u>Time</u> !]	
expiredAtNotIn	[ <u>Time</u> !]	
expiredAtGT	<u>Time</u>	
expiredAtGTE	<u>Time</u>	
expiredAtLT	<u>Time</u>	
expiredAtLTE	<u>Time</u>	
expiredAtIsNil	Boolean	
expiredAtNotNil	Boolean	
createdAt	<u>Time</u>	created_at field predicates
createdAtNEQ	<u>Time</u>	
createdAtIn	[ <u>Time</u> !]	
createdAtNotIn	[ <u>Time</u> !]	
createdAtGT	<u>Time</u>	
createdAtGTE	<u>Time</u>	
createdAtLT	<u>Time</u>	
createdAtLTE	<u>Time</u>	
updatedAt	<u>Time</u>	updated_at field predicates
updatedAtNEQ	<u>Time</u>	
updatedAtIn	[ <u>Time</u> !]	
updatedAtNotIn	[ <u>Time</u> !]	

入力フィールド	型	説明
updatedAtGT	<u>Time</u>	
updatedAtGTE	<u>Time</u>	
updatedAtLT	<u>Time</u>	
updatedAtLTE	<u>Time</u>	

### 3-3-42. CreateEventInput

イベント登録時入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
eventName	String!	イベント名
eventTriggerId	ID!	イベントトリガーID
enabled	Boolean!	イベント有効フラグ
notificationEvent	<u>CreateNotificationEventInput</u>	イベント発生時 通知先設定
writeRemoteOutputEvent	<u>CreateWriteRemoteOutputEventInput</u>	イベント発生時 デバイス制御設定

### 3-3-43. UpdateEventInput

イベント更新時入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
eventName	String	イベント名
eventTriggerId	ID	イベントトリガーID
enabled	Boolean	イベント有効フラグ
notificationEvent	<u>CreateNotificationEventInput</u>	イベント発生時 通知先設定
clearNotificationEvent	Boolean	イベント発生時 通知先設定
writeRemoteOutputEvent	<u>UpdateWriteRemoteOutputEventInput</u>	イベント発生時 デバイス制御設定
clearwriteRemoteOutputEvent	Boolean	

### 3-3-44. CreateNotificationEventInput

イベント発生時 通知先設定の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
notificationTargetIdList	[ID!]!	通知先のIDリスト

### 3-3-45. CreateWriteRemoteOutportEventInput

イベント発生時 デバイス制御設定の追加時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
remoteOutportList	[ <u>CreateRemoteOutportInput</u> !]!	デバイス制御設定のリスト

### 3-3-46. UpdateWriteRemoteOutportEventInput

イベント発生時 デバイス制御設定の更新時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
remoteOutportList	[ <u>UpdateRemoteOutportInput</u> !]!	デバイス制御設定のリスト

### 3-3-47. CreateRemoteOutportInput

イベント発生時 デバイス制御設定の追加時の各入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
host	String!	制御するデバイスのホスト名、または、IPアドレス

本機を指定する場合は空文字を指定する || portNo | OutportPortNoEnum ! | 制御するデバイスの出力ポート番号 || authUser | String! | 制御するデバイスのAPIユーザー名

本機を指定する場合は空文字を指定する || authPassword | String! | 制御するデバイスのAPIユーザーのパスワード

本機を指定する場合は空文字を指定する || oneofOperation | OneofWriteOutportOperation ! | 対象デバイスの出力ポートの制御内容 |

### 3-3-48. UpdateRemoteOutportInput

イベント発生時 デバイス制御設定の更新時の各入力データ



型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
id	ID	更新するID
host	String	制御するデバイスのホスト名、または、IPアドレス

本機を指定する場合は空文字を指定する

未指定時は更新対象外 || portNo | OutportPortNoEnum | 制御するデバイスの出力ポート番号

未指定時は更新対象外 || authUser | String | 制御するデバイスのAPIユーザーのパスワード

本機を指定する場合は空文字を指定する

未指定時は更新対象外 || authPassword | String | 制御するデバイスのAPIユーザーのパスワード

本機を指定する場合は空文字を指定する

未指定時は更新対象外 || oneofOperation | OneofWriteOutportOperation | 対象デバイスの出力ポートの制御内容

未指定時は更新対象外 |

### 3-3-49. CreateEventTriggerInput

イベントトリガー登録時入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String!	イベントトリガー名
oneofEventTrigger	<u>OneofEventTrigger</u> !	イベントトリガー

### 3-3-50. UpdateEventTriggerInput

イベントトリガー更新時入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String	イベントトリガー名
oneofEventTrigger	<u>OneofEventTrigger</u>	イベントトリガー

### 3-3-51. OneofEventTrigger

イベントトリガー

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
inportEventTrigger	<u>InportEventTriggerInput</u>	入力ポート イベントトリガー
outportEventTrigger	<u>OutportEventTriggerInput</u>	出力ポート イベントトリガー
scheduleEventTrigger	<u>ScheduleEventTriggerInput</u>	スケジュール イベントトリガー

### 3-3-52. InportEventTriggerInput

入力ポート イベントトリガー

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
oneofCondition	<u>OneofInportEventTriggerCondition!</u>	

### 3-3-53. OneofInportEventTriggerCondition

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
all	<u>InportEventTriggerConditionAll</u>	指定した条件をすべて満たした時のイベント条件
any	<u>InportEventTriggerConditionAny</u>	指定した条件の内、いずれかを満たした時のイベント条件

### 3-3-54. InportEventTriggerConditionAll

指定した条件をすべて満たした時のイベント条件

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
inportState	<u>InportStateEnum!</u>	ポートの状態
inportPortNoList	[ <u>InportPortNoEnum!</u> ]!	ポートの番号

### 3-3-55. InportEventTriggerConditionAny

指定した条件の内、いずれかを満たした時のイベント条件

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
inportState	<u>InportStateEnum</u> !	ポートの状態
inportPortNoList	[ <u>InportPortNoEnum</u> ! ]!	ポートの番号

### 3-3-56. OutportEventTriggerInput

出力ポート イベントトリガー

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
oneofCondition	<u>OneofOutportEventTriggerCondition</u> !	

### 3-3-57. OneofOutportEventTriggerCondition

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
all	<u>OutportEventTriggerConditionAll</u>	指定した条件をすべて満たした時のイベント条件
any	<u>OutportEventTriggerConditionAny</u>	指定した条件の内、いずれかを満たした時のイベント条件

### 3-3-58. OutportEventTriggerConditionAll

指定した条件をすべて満たした時のイベント条件

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
outputState	<u>OutportStateEnum</u> !	ポートの状態
outputPortNoList	[ <u>OutportPortNoEnum</u> ! ]!	ポートの番号

### 3-3-59. OutportEventTriggerConditionAny

指定した条件の内、いずれかを満たした時のイベント条件

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
outportState	<u>OutportStateEnum</u> !	ポートの状態
outportPortNoList	[ <u>OutportPortNoEnum</u> ! ]	ポートの番号

### 3-3-60. ScheduleEventTriggerInput

スケジュール イベントトリガー

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
date	<u>ScheduleEventTriggerDateInput</u> !	
time	<u>ScheduleEventTriggerTimeInput</u> !	

### 3-3-61. ScheduleEventTriggerDateInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
kind	<u>ScheduleEventTriggerDateKindEnum</u> !	
months	<u>ScheduleEventTriggerDateMonthsInput</u> !	
dayOfWeeks	<u>ScheduleEventTriggerDateDayOfWeeksInput</u> !	

### 3-3-62. ScheduleEventTriggerDateMonthsInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
january	Boolean!	
february	Boolean!	
march	Boolean!	
april	Boolean!	
may	Boolean!	
june	Boolean!	
july	Boolean!	

入力フィールド	型	説明
august	Boolean!	
september	Boolean!	
october	Boolean!	
november	Boolean!	
december	Boolean!	

### 3-3-63. ScheduleEventTriggerDateDayOfWeeksInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
monday	Boolean!	
tuesday	Boolean!	
wednesday	Boolean!	
thursday	Boolean!	
friday	Boolean!	
saturday	Boolean!	
sunday	Boolean!	

### 3-3-64. ScheduleEventTriggerTimeInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
start	String!	
repeat	<a href="#">ScheduleEventTriggerTimeRepeatInput</a>	

### 3-3-65. ScheduleEventTriggerTimeRepeatInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
until	String!	

入力フィールド	型	説明
intervalMinutes	Int!	

### 3-3-66. UpdateInportInput

入力ポート設定更新時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String	入力ポート名
chatterThresholdMillisec	Int	入力ポートのチャタリング無視時間(ミリ秒)

### 3-3-67. UpdateNetworkInput

ネットワーク設定時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
hostName	String	ホスト名
oneofNetworkConfig	<a href="#">OneofNetworkConfigInput</a>	ネットワーク設定

### 3-3-68. OneofNetworkConfigInput

ネットワーク設定の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
dhcp	<a href="#">NetworkConfigDhcpInput</a>	DHCP使用時
static	<a href="#">NetworkConfigStaticInput</a>	静的アドレス指定時

### 3-3-69. NetworkConfigDhcpInput

DHCP使用時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
primaryDns	String	プライマリDNSサーバー
「255.255.255.255」形式で指定		
secondaryDns	String	セカンダリDNSサーバー
「255.255.255.255」形式で指定		

### 3-3-70. NetworkConfigStaticInput

静的アドレス指定時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
ipv4Address	String!	IPアドレス
「255.255.255.255」形式で指定		
subnetMask	Int!	サブネットマスク 1～30の整数

1(128.0.0.0) 2(192.0.0.0) … 29(255.255.255.248) 30(255.255.255.252) || gateway | String! | ゲートウェイ「255.255.255.255」形式で指定 || primaryDns | String! | プライマリDNSサーバー「255.255.255.255」形式で指定 || secondaryDns | String | セカンダリDNSサーバー「255.255.255.255」形式で指定 |

### 3-3-71. CreateNotificationTargetInput

通知先設定新規登録時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String!	通知先名
oneofCreateNotificationTarget	<u>OneofCreateNotificationTarget!</u>	通知設定

### 3-3-72. UpdateNotificationTargetInput

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String	
oneofCreateNotificationTarget	<u>OneofCreateNotificationTarget</u>	

### 3-3-73. OneofCreateNotificationTarget

通知設定

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
email	<u>CreateNotificationTargetEmailInput</u>	メール通知設定
webhook	<u>CreateNotificationTargetWebhookInput</u>	WebHook通知設定

### 3-3-74. CreateNotificationTargetEmailInput

メール通知設定入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
fromName	String!	メール送信元 宛名
fromAddress	String!	メール送信元 アドレス
toAddressList	[String!]!	メール送信先 アドレスリスト
smtpEncryption	<u>SMTPEncryptionEnum</u> !	SMTP 暗号化方式
smtpHost	String!	SMTP ホスト名、または、IPアドレス
smtpPort	Int!	SMTP ポート番号
oneofSmtpAuth	<u>OneofSMTPAuthInput</u> !	SMTP 認証情報

### 3-3-75. OneofSMTPAuthInput

SMTP 認証情報

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
none	<u>SMTPAuthNoneInput</u>	SMTP 認証なし
plain	<u>SMTPAuthPlaneInput</u>	SMTP 認証あり

### 3-3-76. SMTPAuthNoneInput

型: 入力オブジェクト

入力フィールド:



入力フィールド	型	説明
_dummy	<u>Dummy</u>	

### 3-3-77. STMPAuthPaneInput

SMTP 認証情報

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
user	String!	SMTP ユーザ名
password	String!	SMTP パスワード

### 3-3-78. CreateNotificationTargetWebhookInput

WebHook通知設定入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
url	String!	URL
signingPassphrase	String	署名用パスフレーズ

### 3-3-79. NotifyInput

通知テスト時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
subject	String	送信タイトル
body	String	送信本文

### 3-3-80. UpdateOutportInput

出力ポート設定更新時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
name	String	出力ポート名
initialStateOnStartup	<u>OutputInitialStateOnStartupEnum</u>	OS起動時の出力ポート状態の設定

### 3-3-81. WriteOutputInput

出力ポート制御時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
oneofOperation	<u>OneofWriteOutportOperation</u> !	制御内容
logMessage	String!	履歴に記録するメッセージ

### 3-3-82. OneofWriteOutportOperation

出力ポート制御時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
on	<u>WriteOutportOperationOn</u>	出力ポートをONにする
off	<u>WriteOutportOperationOff</u>	出力ポートをOFFにする

### 3-3-83. WriteOutportOperationOn

出力ポートをONにする

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
oneshotDurationMillisec	Int	制御後にOFFにするまでの時間(ミリ秒)
未指定時はONを継続する		

### 3-3-84. WriteOutportOperationOff

出力ポートをOFFにする

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
oneshotDurationMillisec	Int	制御後にONにするまでの時間(ミリ秒)
未指定時はOFFを継続する		

### 3-3-85. OnlineFirmwareUpdateInput

オンラインアップデート時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
firmwareUrl	String!	アップデートするファームウェアのダウンロードURL

### 3-3-86. VerifyUserInput

ユーザ検証時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
userId	String!	ユーザID
password	String!	パスワード

### 3-3-87. CreateUserInput

ユーザ新規登録の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
userId	String!	ユーザID
password	String!	パスワード
kind	<u>UserKindEnum</u> !	権限
expiredAt	<u>Time</u>	有効期限
memo	String	メモ

### 3-3-88. UpdateUserInput

ユーザ情報更新時の入力データ

型: 入力オブジェクト

入力フィールド:

入力フィールド	型	説明
userId	String	ユーザID
未指定時は更新対象外		
password	String	パスワード
未指定時は更新対象外		
kind	<u>UserKindEnum</u>	権限
未指定時は更新対象外		
expiredAt	<u>Time</u>	有効期限
未指定時は更新対象外		
clearExpiredAt	Boolean	有効期限をクリアするかどうか
memo	String	メモ
未指定時は更新対象外		

## 3-4. 列挙型

### 3-4-1. CloudKindEnum

クラウドの種別

型: 列挙型

値:

値	説明
AZURE	Azure iot hub
SMARTIO	SMARTIOクラウド
NONE	クラウドを利用していない

### 3-4-2. CloudStateEnum

型: 列挙型

値:

値	説明
NOT_CONFIGURED	未設定
CONNECTING	接続中
CONNECTED	接続済み

### 3-4-3. EventNotificationOrderField

Properties by which EventNotification connections can be ordered.

型: 列挙型

値:

値	説明
CREATED_AT	
UPDATED_AT	

### 3-4-4. EventOrderField

Properties by which Event connections can be ordered.

型: 列挙型

値:

値	説明
NAME	
ENABLED	
CREATED_AT	
UPDATED_AT	

### 3-4-5. EventTriggerInportOrderField

Properties by which EventTriggerInport connections can be ordered.

型: 列挙型

値:

値	説明
CONDITION	
INPORT_STATE	
CREATED_AT	

値	説明
UPDATED_AT	

### 3-4-6. EventTriggerKindEnum

EventTriggerKindEnum is enum for the field event\_trigger\_kind

型: 列挙型

値:

値	説明
INPORT	
OUTPORT	
SCHEDULE	

### 3-4-7. EventTriggerOrderField

Properties by which EventTrigger connections can be ordered.

型: 列挙型

値:

値	説明
NAME	
EVENT_TRIGGER_KIND	
CREATED_AT	
UPDATED_AT	

### 3-4-8. EventTriggerOutportOrderField

Properties by which EventTriggerOutport connections can be ordered.

型: 列挙型

値:

値	説明
CONDITION	
OUTPORT_STATE	
CREATED_AT	
UPDATED_AT	

### 3-4-9. EventTriggerScheduleOrderField

Properties by which EventTriggerSchedule connections can be ordered.

型: 列挙型

値:

値	説明
DateKind	
CREATED_AT	
UPDATED_AT	

### 3-4-10. EventWriteRemoteOutputOrderField

Properties by which EventWriteRemoteOutput connections can be ordered.

型: 列挙型

値:

値	説明
CREATED_AT	
UPDATED_AT	

### 3-4-11. InportEventTriggerConditionEnum

InportEventTriggerConditionEnum is enum for the field condition

型: 列挙型

値:

値	説明
ANY	
ALL	

### 3-4-12. InportEventTriggerInportStateEnum

InportEventTriggerInportStateEnum is enum for the field inport\_state

型: 列挙型

値:

値	説明
ON	

値	説明
OFF	

### 3-4-13. InportOrderField

Properties by which Inport connections can be ordered.

型: 列挙型

値:

値	説明
PORT_NO	
NAME	
CHATTER_THRESHOLD_MILLISEC	
CREATED_AT	
UPDATED_AT	

### 3-4-14. InportPortNoEnum

InportPortNoEnum is enum for the field port\_no

型: 列挙型

値:

値	説明
ONE	
TWO	
THREE	
FOUR	
FIVE	
SIX	
SEVEN	
EIGHT	

### 3-4-15. LogEventOrderField

Properties by which LogEvent connections can be ordered.

型: 列挙型

値:



値	説明
EVENT_ID	
EVENT_TRIGGER_ID	
EVENT_TRIGGER_KIND	
EVENT_TRIGGER_NAME	
EVENT_NAME	
CREATED_AT	
UPDATED_AT	

### 3-4-16. LogInportOrderField

Properties by which LogInport connections can be ordered.

型: 列挙型

値:

値	説明
PORT_NO	
PORT_NAME	
INPUT_VALUE	
CREATED_AT	
UPDATED_AT	

### 3-4-17. LogOutportOrderField

Properties by which LogOutport connections can be ordered.

型: 列挙型

値:

値	説明
PORT_NO	
PORT_NAME	
OPERATION	
ONESHOT_DURATION_MILLISEC	
MESSAGE	

値	説明
CREATED_AT	
UPDATED_AT	

### 3-4-18. NotificationTargetEmailOrderField

Properties by which NotificationTargetEmail connections can be ordered.

型: 列挙型

値:

値	説明
CREATED_AT	
UPDATED_AT	

### 3-4-19. NotificationTargetKindEnum

NotificationTargetKindEnum is enum for the field notification\_target\_kind

型: 列挙型

値:

値	説明
EMAIL	
WEBHOOK	

### 3-4-20. NotificationTargetOrderField

Properties by which NotificationTarget connections can be ordered.

型: 列挙型

値:

値	説明
NAME	
NOTIFICATION_TARGET_KIND	
CREATED_AT	
UPDATED_AT	

### 3-4-21. NotificationTargetWebhookOrderField

Properties by which NotificationTargetWebhook connections can be ordered.

型: 列挙型

値:

値	説明
CREATED_AT	
UPDATED_AT	

### 3-4-22. OrderDirection

Possible directions in which to order a list of items when provided an `orderBy` argument.

型: 列挙型

値:

値	説明
ASC	Specifies an ascending order for a given <code>orderBy</code> argument.
DESC	Specifies a descending order for a given <code>orderBy</code> argument.

### 3-4-23. OutportEventTriggerConditionEnum

OutportEventTriggerConditionEnum is enum for the field condition

型: 列挙型

値:

値	説明
ANY	
ALL	

### 3-4-24. OutportEventTriggerOutportStateEnum

OutportEventTriggerOutportStateEnum is enum for the field outport\_state

型: 列挙型

値:

値	説明
ON	
OFF	

### 3-4-25. OutportInitialStateOnStartupEnum

OutportInitialStateOnStartupEnum is enum for the field initial\_state\_on\_startup

型: 列挙型

値:

値	説明
ON	
OFF	

### 3-4-26. OutportOrderField

Properties by which Outport connections can be ordered.

型: 列挙型

値:

値	説明
PORT_NO	
NAME	
INITIAL_STATE_ON_STARTUP	
CREATED_AT	
UPDATED_AT	

### 3-4-27. OutportPortNoEnum

OutportPortNoEnum is enum for the field port\_no

型: 列挙型

値:

値	説明
ONE	
TWO	
THREE	
FOUR	
FIVE	
SIX	
SEVEN	
EIGHT	

### 3-4-28. RemoteOutportOperationEnum

RemoteOutportOperationEnum is enum for the field operation

型: 列挙型

値:

値	説明
OUTPORT_ON	
OUTPORT_OFF	

### 3-4-29. RemoteOutportOrderField

Properties by which RemoteOutport connections can be ordered.

型: 列挙型

値:

値	説明
HOST	
PORT_NO	
AUTH_USER	
operation	
ONESHOT_DURATION_MILLISEC	
CREATED_AT	
UPDATED_AT	

### 3-4-30. SMTPAuthEnum

SMTPAuthEnum is enum for the field smtp\_auth

型: 列挙型

値:

値	説明
NONE	
PLAIN	

### 3-4-31. SMTPEncryptionEnum

SMTPEncryptionEnum is enum for the field smtp\_encryption

型: 列挙型

値:

値	説明
NONE	
SMTPS	
STARTTLS	

### 3-4-32. ScheduleEventTriggerDateKindEnum

ScheduleEventTriggerDateKindEnum is enum for the field date\_kind

型: 列挙型

値:

値	説明
DAILY	
WEEKLY	
YEARLY	

### 3-4-33. UserKindEnum

UserKindEnum is enum for the field user\_kind

型: 列挙型

値:

値	説明
DEVICE_ADMIN	
DEVICE_USER	
API_CONSUMER	

### 3-4-34. UserOrderField

Properties by which User connections can be ordered.

型: 列挙型

値:

値	説明
USER_ID	
USER_KIND	

値	説明
MEMO	
EXPIRED_AT	
CREATED_AT	
UPDATED_AT	

### 3-4-35. InportStateEnum

型: 列挙型

値:

値	説明
ON	
OFF	

### 3-4-36. OutportStateEnum

型: 列挙型

値:

値	説明
ON	
OFF	

## 3-5. スカラー型

### 3-5-1. ID

The `ID` scalar type represents a unique identifier, often used to refetch an object or as key for a cache. The ID type appears in a JSON response as a String; however, it is not intended to be human-readable. When expected as an input type, any string (such as `"4"`) or integer (such as `4`) input value will be accepted as an ID.

型: スカラー

### 3-5-2. Time

RFC3339Nano format string `"2006-04-13T14:12:53.4242+09:00"`

型: スカラー

### 3-5-3. Map

型: スカラー

### 3-5-4. Dummy

空のtype/inputを定義するのに使用する

型: スカラー

### 3-5-5. String

The `String` scalar type represents textual data, represented as UTF-8 character sequences. The `String` type is most often used by GraphQL to represent free-form human-readable text.

型: スカラー

### 3-5-6. Int

The `Int` scalar type represents non-fractional signed whole numeric values. `Int` can represent values between  $-(2^{31})$  and  $2^{31} - 1$ .

型: スカラー

### 3-5-7. Cursor

Define a Relay Cursor type: <https://relay.dev/graphql/connections.htm#sec-Cursor>

型: スカラー

### 3-5-8. Boolean

The `Boolean` scalar type represents `true` or `false`.

型: スカラー