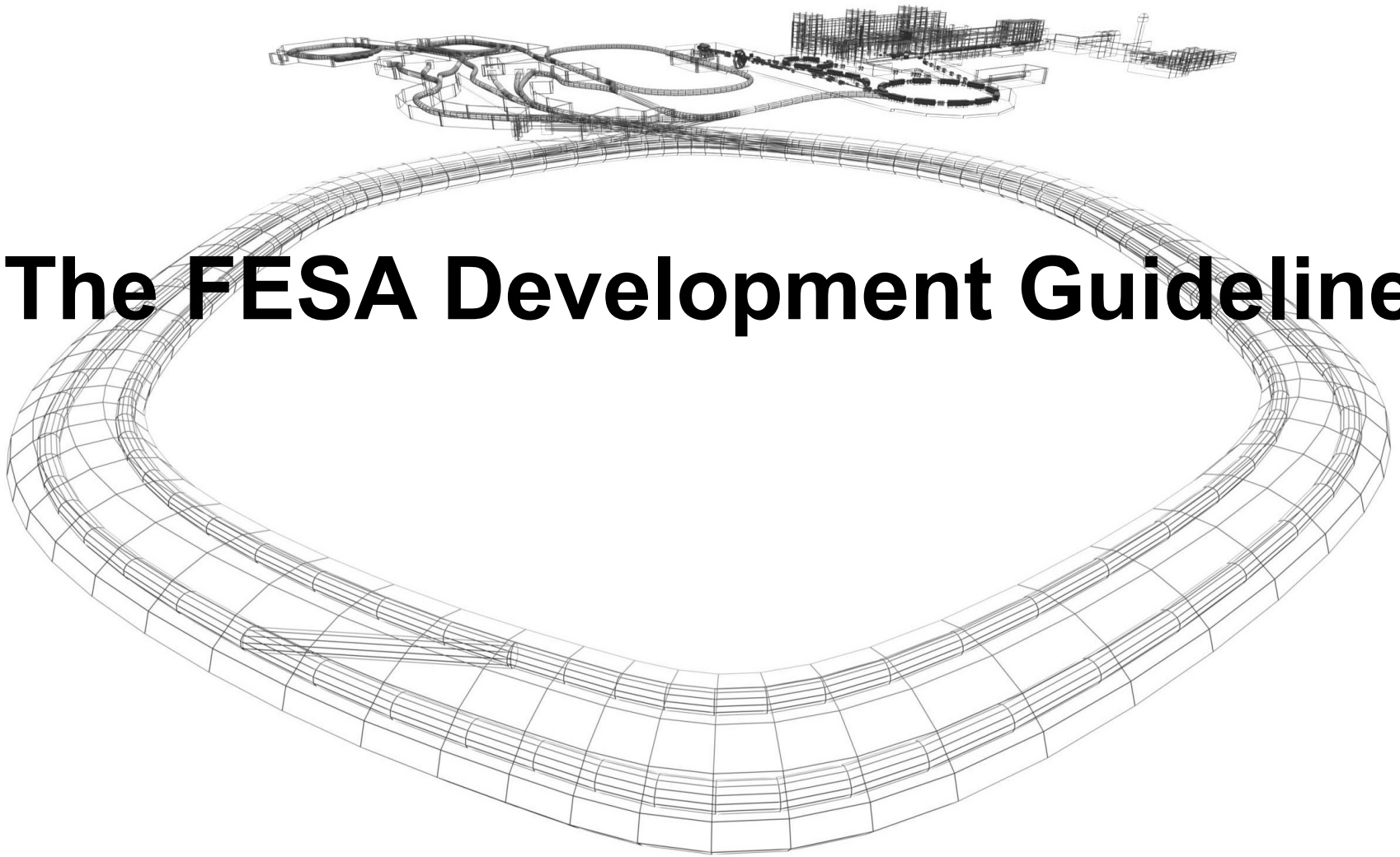
The FESA Development Guideline



Topics

- **Overview**
- Properties
- Fields
- Misc

Overview

- Why we need a GSI – Convention ?
- Wiki: “development workflow for productive classes”
  - Read the “FESA Development Guideline” !!
  - CSCO-AP
  - Mocking (possibility to simulate hardware)
- “Add Class” → GSIClassTemplate
- GSI-specific-elements will be enforced for productive classes !
- Use the FESA-browser for detailed information about the xml-elements.

Topics

- Overview
- **Properties**
- Fields
- Misc

Properties

GSI-Status-Property	Status, acqStamp, updateFlags, cycleName, cycleStamp, status, det
multiplexed	false
name	Status
on-change	true
subscribable	true
visibility	operational
acq-stamp-item	acqStamp, int64_t
update-flag-item	updateFlags,
cycle-name-item	1D array: cycleName, char
cycle-stamp-item	cycleStamp, int64_t
get-action	(server-action-ref | abstract-server-action)
status-item	status,
detailed-status-item	1D array: detailedStatus, bool,
detailed-status-labels-item	2D array: detailedStatus_labels, char,
detailed-status-severity-item	detailedStatus_severity,
powerState-item	powerState,
control-item	control,
interlock-item	interlock, bool,
opReady-item	opReady, bool,
error_collection-item	error_codes, error_messages, error_timestamps, error_cycle_names,

Properties

GSI-Acquisition-Property	(((description*), (
multiplexed	true
name	Acquisition
on-change	true
subscribable	true
visibility	operational
value-item	((description*, (s
acq-stamp-item	(description*, sca
update-flag-item	(description*, cus
cycle-name-item	(description*, arr
cycle-stamp-item	(description*, sca
get-action	(server-action-ref
multiplexing-context-item	((acqStamp, cycl
direction	OUT
acqStamp	(scalar)
cycleStamp	(scalar)
cycleName	(array)
multiplexing-context-field-ref	

Properties

GSI-Version-Property	(((description*), (filter-ite
multiplexed	false
name	Version
on-change	false
subscribable	false
visibility	operational
acq-stamp-item	(description*, scalar, dat
update-flag-item	(description*, custom-ty
cycle-name-item	(description*, array, data
cycle-stamp-item	(description*, scalar, dat
get-action	(server-action-ref | abstr
version-item	(array)
direction	OUT
name	deploy_unit_version
array	(custom-constant-dim)
version-item	(array)
direction	OUT
name	fesa_version
array	(custom-constant-dim)

Topics

- Overview
- Properties
- **Fields**
- Misc

Fields

data	(device-data?, global-data?, tim
device-data	(configuration?, setting?, acqui
configuration	detailedStatus_labels, detailedS
GSI-detailed-status-labels-field	2D array: detailedStatus_labels,
GSI-detailed-status-severity-field	detailedStatus_severity,
setting	dummyField, power
field	dummyField, bool
GSI-power-field	power,
acquisition	control, powerState, status, int
GSI-control-field	control,
GSI-powerState-field	powerState,
GSI-status-field	status,
GSI-interlock-field	interlock, bool
GSI-opReady-field	opReady, bool
GSI-detailed-status-field	1D array: detailedStatus, bool
GSI-multiplexing-context-field	acquisitionContext,
GSI-error_collection-field	error_collection,

Fields (Status-Property)

GSI-status-field	(custom-type-scal
multiplexed	false
name	status
custom-type-scalar	
data-type-name-ref	DEVICE_STATUS

Custom-type: DEVICE_STATUS:
Enum(0=UNKNOWN, 1=OK, 2=WARNING, 3=ERROR)
Overall-Status of the device

Fields (Status-Property)

acquisition	(((fault-field?, state-field?,
GSI-detailed-status-field	(array)
multiplexed	false
name	detailedStatus

configuration	((hw-address?, device-relations-field?, (fie
GSI-detailed-status-labels-field	(array2D, default)
name	detailedStatus_labels

configuration	detailedStatus_labels, detailedStatus_severity
GSI-detailed-status-labels-field	2D array: detailedStatus_labels, char,
GSI-detailed-status-severity-field	detailedStatus_severity,
name	detailedStatus_severity
custom-type-array	ref: DETAILED_STATUS_SEVERITY
data-type-name-ref	DETAILED_STATUS_SEVERITY
custom-constant-dim	ref: DETAILED_STATUS_SIZE
constant-name-ref	DETAILED_STATUS_SIZE
default	{INFO,INFO}

Custom-type: DETAILED_STATUS_SEVERITY:
Enum(0=INFO, 1=OK, 2=WARNING_ON_FALSE, 3=ERROR_ON_FALSE)

Fields (Power-Property)

setting	(((state-field?, cycle
field	(description*, (scala
GSI-power-field	(custom-type-scalar
multiplexed	false
name	power
persistent	true
custom-type-scalar	
data-type-name-ref	DEVICE_POWER

Custom-type: DEVICE_POWER
Enum (1=ON, 2=OFF)
Desired power-state of the device

Fields (Status-Property)

GSI-powerState-field	(custom-type-scalar)
multiplexed	false
name	powerState
custom-type-scalar	
data-type-name-ref	DEVICE_POWER_STATE

Custom-type: DEVICE_POWER_STATE
Enum (0=UNKNOWN, 1=ON, 2=OFF, 3=STANDBY, 4=POWER_DOWN, 5=POWER_UP)
power-state of the device

Fields (Status-Property)

GSI-control-field	(custom-type-scal
multiplexed	false
name	control
custom-type-scalar	
data-type-name-ref	DEVICE_CONTROL

Custom-type: DEVICE_CONTROL
Enum (0=REMOTE, 1=LOCAL)
Device is controlled locally, or via the control-system ?

Fields (Status-Property)

GSI-interlock-field	(scalar)
multiplexed	false
name	interlock
scalar	
type	bool

GSI-opReady-field	(scalar)
multiplexed	false
name	opReady
scalar	
type	bool

Fields (Status-Property)

GSI-error_collection-field	(custom-type-array)
multiplexed	false
name	error_collection
custom-type-array	(custom-constant-dim)
data-type-name-ref	GSI_ERROR
custom-constant-dim	
constant-name-ref	MAX_NUMBER_OF_ERROR_MESSAGES

custom-types	(notification-
struct	(description*,
name	GSI_ERROR
struct-item	(description*, (scal
name	error_string
array	((dim | custom-con
struct-item	(description*, (scal
name	error_code
scalar	
struct-item	(description*, (scal
name	error_timestamp
scalar	
struct-item	(description*, (scal
name	error_cycle_name
array	((dim | custom-con

- store last errors for the clients
- log error to central logging system

```
std::string errorString= "TestError";
long error_code= 4711;
(*iter)->error_collection.addError(error_code,errorString,context,(*iter));
```

Fields (Acquisition-Property)

GSI-multiplexing-context-field	(custom-type-scalar)
multiplexed	true
name	acquisitionContext
custom-type-scalar	
data-type-name-ref	GSI_MUX_CONTEX

struct	(description*, struct-ite
name	GSI_MUX_CONTEXT
struct-item	(description*, (scalar |
name	acqStamp
scalar	
struct-item	(description*, (scalar |
name	cycleStamp
scalar	
struct-item	(description*, (scalar |
name	cycleName
array	((dim | custom-constar

```
// easy method
(*device)->acquisitionContext.insert(pCtxt);

// advanced method ( usage of self-defined acquisition-stamp )
// stamp in Nanoseconds  UTC
int64_t stamp = 12345678;
(*device)->acquisitionContext.insert(pCtxt,stamp);
```

Topics

- Overview
- Properties
- Fields
- **Misc**

Misc

Meta-data - field suffixes to use:

- _status
- _min / _max
- _toAbs
- _ tolRel
- _tolCheckMode
- _acqStamp
- _unit

Quality of Acq. Data

- NOT_OK
- BAD_QUALITY
- DIFFERENT_FROM_SETTING
- OUT_OF_RANGE
- BUSY
- TIMEOUT
- …

… and many more standards which should be used at GSI !

For details, check the FESA-Development-Guideline in the Wiki !!

Thanks for your attention!