

Working with Probes

Table of contents

1 Choose a Probe type	1
2 Create and update a Probe	1
3 Start from a portable specification	2
4 Inspect and remove Probes	3

Probes describe where Coreform should evaluate selected output variables. They are persistent Abaqus records: they are saved with the model database, restored when it is reopened, and journaled when changed through the public API.

1 Choose a Probe type

Use a **point Probe** for one location, a **line Probe** for evenly spaced locations along a segment, and a **multi-point Probe** for an explicit collection of unrelated locations.

Each Probe targets exactly one kind of Abaqus repository:

- `instanceNames` selects assembly instances.
- `setNames` selects assembly sets.
- `partNames` selects model parts.

Choose the target whose geometry contains the evaluation locations. Supplying more than one target kind is invalid.

2 Create and update a Probe

The following script creates a line Probe, changes its target and sampling density, then inspects its portable specification:

```
from abaqus import mdb
import coreform.abaqus

coreform.abaqus.init_iga_mdb()

model = mdb.models["Beam"]
probes = model.steps.customData.CoreformIGA

probe = probes.LineProbe(
    name="Centerline",
    instanceNames=("BEAM-1",),
```

```
variables=("S", "U"),
startLocation=(0.0, 0.0, 0.0),
stopLocation=(10.0, 0.0, 0.0),
numEvalPoints=41,
)

probe.setValues(
    setNames=("CENTERLINE",),
    numEvalPoints=81,
)

spec = probe.toSpec()
print(spec.start_location)
print(spec.stop_location)
print(spec.num_eval_points)
```

Passing one target keyword to `setValues()` switches the target kind. Other omitted values retain their current state.

3 Start from a portable specification

Use `spec=` when a portable Probe is the single source of truth. Do not mix `spec=` with the flat Abaqus definition parameters.

```
from coreform.probe import MultiPointProbeSpec, ProbeRegion

portable = MultiPointProbeSpec(
    name="Fasteners",
    region=ProbeRegion.sets(("FASTENER-REGION",)),
    variables=("S",),
    locations=(
        (1.0, 0.0, 0.0),
        (3.0, 0.0, 0.0),
        (5.0, 0.0, 0.0),
    ),
)

probe = probes.MultiPointProbe(spec=portable)
```

A mapping produced by a compatible specification is also accepted through `spec=` and is validated before the record is created.

4 Inspect and remove Probes

Probe records are stored by name in `probes.probes`. Resolved `instances`, `sets`, and `parts` properties provide the live Abaqus objects for the active target kind. Call `validate()` after changing surrounding Abaqus model data to check that the stored target names still resolve.

Delete a Probe through its namespace so the operation is journaled:

```
probes.deleteProbe(name="Centerline")
```

See the generated [Abaqus Probe reference](#) and [portable Probe reference](#) for every factory, record attribute, method, portable class, parameter, and return value.